

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ  
імені ІГОРЯ СІКОРСЬКОГО»

# ІНФОРМАЦІЙНІ ТЕХНОЛОГІЇ В ПРИЛАДОБУДУВАННІ ПРАКТИКУМ

**Навчальний посібник**

Рекомендовано Методичною радою КПІ ім. Ігоря Сікорського  
як навчальний посібник для здобувачів ступеня бакалавра  
за освітньою програмою «Автоматизація та комп'ютерно-інтегровані технології в  
приладобудуванні»  
спеціальності 151 «Автоматизація та комп'ютерно-інтегровані технології»

Укладачі: І. В. Кравченко, М.С. Мамута

Електронне мережне навчальне видання

Київ  
КПІ ім. Ігоря Сікорського  
2022

Рецензент: *Мельник Оксана Миколаївна*, канд. пед. наук, старший дослідник, завідувач сектору науково-методичного розвитку інформаційно-цифрової компетентності відділу цифрових освітньо-наукових систем Державної наукової установи «Інститут модернізації змісту освіти»

*Кагляк Олексій Дмитрович*, канд. техн. наук, доцент, в.о. завідувача кафедри лазерної техніки та фізико-технічних технологій інституту матеріалознавства та зварювання ім. Є.О. Патона КПІ ім. Ігоря Сікорського

Відповідальний редактор *Колобродов Валентин Георгійович*, док. техн. наук, професор кафедри комп'ютерно-інтегрованих оптичних та навігаційних систем КПІ ім. Ігоря Сікорського

*Гриф надано Методичною радою КПІ ім. Ігоря Сікорського  
(протокол № 4 від 07.04.2022 р.)  
за поданням Вченої ради приладобудівного факультету  
(протокол № 1/22 від 31.01.2022 р.)*

Містить теоретичний матеріал, приклади та комплекс завдань для комп'ютерного практикуму та самостійної роботи з дисципліни "Інформаційні технології в приладобудуванні". Розглянуто питання чисельних та символічних розрахунків, програмування алгоритмів користувача, роботи з матрицями, графічного виведення результатів, файлового обміну в системах комп'ютерної математики та оформлення науково-технічної документації в офісних застосунках. Викладено відомості щодо технології розв'язання математичних та технічних задач в СКМ «MATHCAD», «MATLAB», оформлення звітів, статей, проєктів тощо засобами Microsoft Office згідно вимог нормативних документів України. Може бути корисний студентам та фахівцям технічних спеціальностей для набуття навичок застосування інформаційних технологій у вигляді сучасних СКМ в учбовій та науково-технічній практиці.

Реєстр. № П 21/22-326. Обсяг 14.6 авт. арк.

Національний технічний університет України  
«Київський політехнічний інститут імені Ігоря Сікорського»  
проспект Перемоги, 37, м. Київ, 03056  
<https://kpi.ua>

Свідоцтво про внесення до Державного реєстру видавців, виготовлювачів і розповсюджувачів видавничої продукції ДК № 5354 від 25.05.2017 р.

© І. В. Кравченко, М. С. Мамаута

© КПІ ім. Ігоря Сікорського, 2022

## ЗМІСТ

ВСТУП .....	6
1. ОФІСНІ СИСТЕМИ.....	7
Практикум 1.1. Оформлення текстових документів в MS Word.....	9
1.1.1. Теоретичні положення.....	9
1.1.2. Завдання для самостійного виконання .....	16
Практикум 1.2. Оформлення таблиць та зображень в MS Word.....	21
1.2.1. Теоретичні положення.....	21
1.2.2. Завдання для самостійного виконання .....	22
Практикум 1.3. Оформлення науково-технічних матеріалів в MS Word .....	23
1.3.1. Теоретичні положення.....	23
1.3.2. Завдання для виконання .....	27
Практикум 2.1. Форматування таблиць в MS Excel .....	30
2.1.1. Теоретичні положення.....	30
2.1.2. Завдання для самостійного виконання .....	36
Практикум 2.2. Обмін даними в MS Excel.....	38
2.2.1. Теоретичні положення.....	38
2.2.2. Завдання для самостійного виконання .....	40
Практикум 2.3. Макроси MS Excel.....	41
2.3.1. Теоретичні положення.....	41
2.3.2. Завдання для самостійного виконання .....	45
3. СКМ MATHCAD.....	47
Практикум 3.1. Загальні обчислення.....	53
3.1.1. Теоретичні положення.....	53
3.1.2. Приклади та завдання для самостійного виконання .....	81
Практикум 3.2. Програмування .....	94

3.2.1. Теоретичні положення.....	94
3.2.2. Завдання для самостійного виконання .....	106
Практикум 3.3. Двомірні графіки .....	107
3.3.1. Теоретичні положення.....	107
3.3.2. Приклади та завдання для самостійного виконання .....	118
Практикум 3.4. Тримірні графіки .....	121
3.4.1. Теоретичні положення.....	121
3.4.2. Приклади та завдання для самостійного виконання .....	137
Практикум 3.5. Розрахунки з розмірностями .....	142
3.5.1. Теоретичні положення.....	142
3.5.2. Приклади та завдання для самостійного виконання .....	148
Практикум 3.6. Символьні перетворення .....	153
3.6.2. Теоретичні положення.....	153
3.6.1. Завдання для самостійного виконання .....	169
Практикум 3.7. Обмін даними.....	170
3.7.1. Теоретичні положення.....	170
3.7.2. Завдання для самостійного виконання .....	186
4. СКМ MATLAB .....	188
Практикум 4.1. Розрахунки в режимі калькулятора.....	192
4.1.1. Теоретичні положення.....	192
4.1.2. Завдання для самостійного виконання .....	209
Практикум 4.2. Програмування .....	211
4.2.1. Теоретичні положення.....	211
4.2.2. Завдання для самостійного виконання .....	224
Практикум 4.3. Двомірні графіки .....	227
4.3.1. Теоретичні положення.....	227
4.3.2. Завдання для самостійного виконання .....	241

Практикум 4.4. Тримірні графіки .....	244
4.4.1. Теоретичні положення.....	244
4.4.2. Завдання для самостійного виконання .....	249
Практикум 4.5. Символьні обчислення.....	250
4.5.1. Засоби Matlab.....	250
4.6.2. Засоби MuPAD .....	256
4.5.3. Завдання для самостійного виконання .....	261
Практикум 4.6. Експорт-імпорт даних .....	263
4.6.1. Теоретичні положення.....	263
4.6.2. Завдання для самостійного виконання .....	275
Література .....	277
Додаток А. Основні функції Matlab .....	279

## ВСТУП

Навчальний посібник призначено для забезпечення інформаційними та методичними матеріалами дисципліни «Інформаційні технології в приладобудуванні».

Дисципліна «Інформаційні технології в приладобудуванні» викладається на першому курсі навчання та має обсяг в 36 годин лекцій та 54 години комп'ютерного практикума. Тому видання не торкається питань чисельних методів, методів оптимізації, обробки експериментальних даних. Увага приділена базовим можливостям для початку роботи з комп'ютерними системами математики та оформленню текстової науково-технічної документації відповідно до вимог нормативних документів України, зокрема ДСТУ 3008:2015, ДСТУ 7.1:2006, ДСТУ 8302:2015, ГОСТ 2.106-96, студентами початкового курсу.

В посібнику викладено необхідні відомості щодо особливостей застосування, технології розв'язання математичних та технічних задач, питання чисельних та символічних розрахунків, програмування алгоритмів користувача, роботи з матрицями, графічного виведення результатів, файлового обміну в системах комп'ютерної математики (СКМ) *MathCAD*, *Matlab* та форматування стилів тексту та аркуша, розрахунків в таблицях, правил цитування для оформлення науково-технічної документації в офісних застосунках в пакеті *Microsoft Office*. Матеріал містить практичні приклади та завдання для самостійної роботи студентів.

Приклади та завдання в посібнику орієнтовані на останню «класичну» версію *MathCAD 15* та версію *MathCAD Prime 7X*, *Matlab 8X*, *Microsoft Office 2016*. Види вікон діалогу та синтаксис команд в інших версіях пакетів можуть дещо відрізнятися від наведених у даному посібнику.

Посібник є розширеним, виправленим та оновленим виданням авторів 2018 року [1].

Мета навчального посібника – допомога студентам в самостійному вивченні відповідних розділів навчальної дисципліни, в набутті навичок застосування інформаційних технологій у вигляді сучасної СКМ та офісного пакета в учбовій та науково-технічній практиці. Посібник може використовуватись для самостійної роботи та дистанційного навчання.

# 1. ОФІСНІ СИСТЕМИ

## Робоче вікно редактора

В робочому вікні редактора (рис. 1.1) розміщуються органи керування та робоче поле 1. Керування редактором проводиться з панелі швидкого доступу (Quick Access Toolbar) 2, стрічки (Ribbon) 3, рядка стану (Status bar) 4.

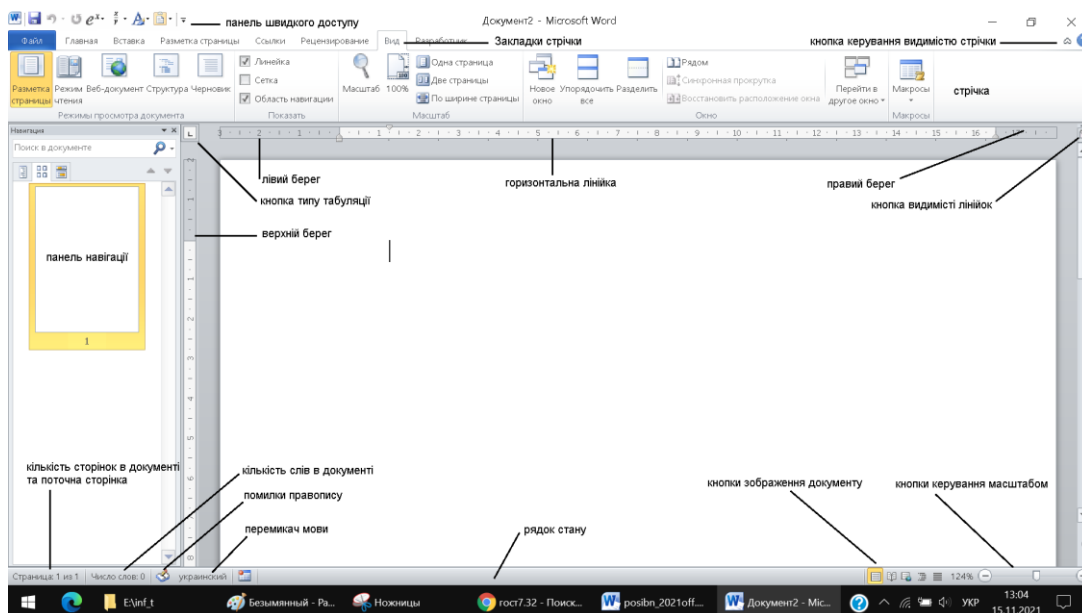


Рис. 1.1. Робоче вікно редактора

Стрічка є головним засобом керування редактором. Замінити стрічку на панель чи меню неможливо. На стрічці елементи керування згруповані на закладках (Tabs) 5 відповідно до виконуваних дій. Згорнути/розгорнути стрічку можна «гарячими» клавішами *CTRL+F1*, або пунктами кнопки контролю видимості, контекстного меню стрічки.

Закладка **File** стрічки відкриває вікно властивостей (*Backstage view*). Вікно розбито на дві частини. Зліва знаходиться стовпець кнопок основних дій з файлом. Кнопка **Option** в ньому слугує для налаштувань редактора.

Більшість елементів керування та виділені в документі фрагменти мають власні контекстне меню (*Shortcut menu*). Відкривається контекстне меню правим щикликом миші над обраним елементом.

Групи закладок стрічки мають **Dialog box launcher** – стрілку в нижньому правому куті на панелі (рис. 1.1). Такі позначки слугують для відкриття діалогових вікон налаштування елементів.

При наведенні миші на елемент керування на екрані автоматично з'являється спливаюче інформаційне вікно (*ScreenTip*) з коротким описом дії елементу.

Рядок стану знаходиться внизу робочого вікна. В ньому відображається загальна інформація про документ та містяться деякі елементи керування зображенням документу. Типовою конфігурацією рядку стану є поле інформації про кількість та номер поточної сторінки **Страница (Page)** та слів **Слов (Words)** документу, містяться позначка помилок правопису, кнопка визначення мови документу, кнопки зображення документу (**View controls**) кнопки керування масштабом зображення документу (**Zoom controls**). Склад елементів рядка статусу можна змінювати в контекстному меню рядка.

На панелі швидкого доступу розміщуються кнопки команд, які виконуються найчастіше. Зазвичай це є команди збереження документу, відміни останніх дій, повторення останньої дії. Користувач має змогу змінити склад кнопок на панелі швидкого доступу.

На робочому полі редактора зображується сторінка документу. Документ може бути зображений в режимі читання *Read More*, веб сторінки *Web Layout*, розмітки сторінки *Print Layout*, показу структури складеного документу *Outline*, чернетки *Draft*.


Найбільш зручним для роботи є режим розмітки сторінки. Перемикання між режимами зображення документу проводиться кнопками панелі стану (рис. 1.1) або кнопками **View-Views** стрічки.

Зазвичай сторінки відокремлюються між собою смужкою фону сірого кольору (*white space*). Керування видимістю смужки проводиться подвійним щикликом на зоні на маркері двох стрілок (*up-down arrow*).

Для спрощення вирівнювання елементів на сторінці документа може застосовуватися сітка (*Gridlines*) та горизонтальна та вертикальна лінійки (*Rulers*) та відображення недрукованих символів (*Nonprinting characters*).



Горизонтальна лінійка розміщується над сторінкою. На ній відмічаються мітки параметрів абзацу, берегів, табуляції. Вертикальна лінійка розміщується зліва від сторінки. На ній відмічаються береги.

Лінійки відображаються в режимі розмітки сторінки та веб документів. Перемикання видимості лінійок та сітки проводиться пунктом **View-Show** стрічки або кнопкою , яка знаходиться в правому верхньому куті сторінки над вертикальним повзунком.

В меню **File-Options-Advanced-Display** можна задати видимість вертикальної лінійки **Show vertical ruler...** та одиниці виміру лінійки **Show measurements in units of**.

Параметри сітки визначаються у вікні *Gridlines Navigation Pane*. Вікно відкривається пунктом **Layout-Align-Gridlines options**.

Зліва від аркушу документа можна вивести пунктом **View-Show-Navigation pane** стрічки панель навігації (рис. 1.1). На панелі відображається список розділів документа або сторінки документа. Простим кліком по обраному елементу панелі проводиться виведення відповідного фрагменту документа на екран.

При виділенні фрагменту тексту над ним може автоматично з'явитися вікно (*Mini toolbar*), яке містить основні дії форматування. З'являється автоматично після виділення тексту. Прибрати вікно можна пунктом **File-Options-General-Show mini toolbar on selection** стрічки.

## Практикум 1.1. Оформлення текстових документів в MS Word

### 1.1.1. Теоретичні положення

Символи тексту документу вводяться за виглядом контуру малювання різними шрифтами. Розрізняють *пропорційний* тип шрифтів та *моно* тип. В пропорційних шрифтах ширина символів змінюється, в моно шрифтах ширина всіх символів є сталою. В свою чергу пропорційні шрифти розділяють на родину *Serif*, в яких на кінцях ліній символів є декоративні засічки, та родину *San Serif*, контури символів яких не мають декоративних засічок.

Конкретний набір символів з відповідним способом малювання контуру називають називають *гарнітурою* або просто шрифтом по імені. Наприклад, пропорційний шрифт родини *Serif*, тобто з декорацією контуру, *Times New Roman*, шрифт гарнітури *Times New Roman* або просто шрифт *Times New Roman*. Пропорційний шрифт родини *San Serif Arial*, моношрифт *Monotxt*.

Для шрифтів в документі визначається розмір або кегль, колір, стиль написання тощо.

*Кегль* – одиниця виміру розміру шрифтів, який визначається в «точках» (*point size*), 1pt=1/72 дюйма [2], тобто символ кеглем 14 має висоту 5 мм.

Параметри поточного шрифту визначаються полями *Main-Fonts* стрічки або у вікні шрифтів.

Шрифтом за ГОСТ 2.304-81 (ДСТУ ISO 3098:2006) є *isocpueer.ttf*.

Для спрощення табличного розташування слів по стовпцях в кількох рядках без створення таблиці застосовується табуляція. Табуляцією слова відокремлюються не клавішею пробілу на один символ, а клавішею табуляції на крок табуляції. За замовчання крок табуляції є 1.27 см.

Табульовані елементи можуть бути вирівняні відносно позначки табуляції з лівого боку, з правого боку, по центру тощо.

Тип вирівнювання визначається кнопкою у верхньому лівому куті сторінки над вертикальною лінійкою. Перемикання типів проводиться послідовними кліками на кнопки.

Створення табульованого рядка може проводитися декількома способами. Можна спочатку розмітити табульований рядок, а потім ввести в нього слова. Введення слів починається з натискання клавіши табуляції. Можна спочатку набрати слова, провести розмітку рядка, а потім зліва направо ставити курсор на початок слова та натискати клавішу табуляції.

Переведення на наступний рядок зберігає розмітку табуляції.

Розмітка рядка проводиться парами дій. Вибір типу вирівнювання-клік на обраному місці горизонтальної лінійки. На лінійці залишаються позначки обраного типу у відповідних місцях. Для розмітки та редагування табуляції призначено вікно *Tabs* (рис. 1.2).

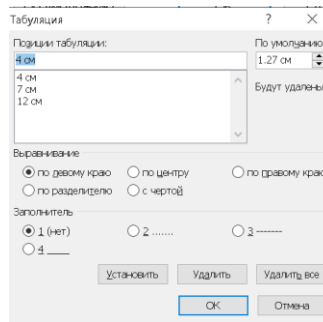



Рис. 1.2. Табуляція

Викликається вікно кнопкою **Tab**s... у вікні налаштування абзаців.

Більшість символів, які є найбільш вживаними, в документах вводяться однократним натискання відповідних клавіш клавіатури. Менш вживані символи можна ввести з вікна вставляння символів. Вікно активується

кнопкою  стрічки. Корисні для науково-технічних документів символи знаходяться в наборах грецькі та коптські символи, знаки пунктуації, стрілки, математичні оператори, технічні знаки.

Будь-який символ може бути вставлений за своїм кодом. Знак Юнікоду за шістнадцатеричним кодом вставляється комбінацією клавіш *чотиризначний шістнадцятиричний код*+ALT+X . Наприклад, символ «€» може бути вставлений комбінацією 20AC+ ALT+X.

Знак ANSI за десятковим кодом вставляється комбінацією клавіш ALT +тризначний десятковий код на цифровій клавіатурі. Наприклад, символ «@» , може бути вставлений комбінацією ALT+64(цифровий).

Редактор має кілька різновидів символу тире: дефіс «-», тире «—», довге тире «—». Український правопис [3] передбачає застосування дефісу в складених словах та тире – в інших випадках.

Символ тире вводиться комбінацією клавіш *Ctrl*+-. Символ «-» вводиться на цифровій клавіатурі.

За умови активного режиму автозаміни символ дефісу автоматично перетворюється на символ тире, якщо дефіс вводити в наступній послідовності: слово – пробіл – дефіс – пробіл – слово – пробіл.

У випадках, коли необхідно забезпечити нерозривність послідовних слів або складного слова застосовують нерозривний пробіл та нерозривний

дефіс. Наприклад, нерозривний пробіл між прізвищем та ініціалами – «Сірий Іван», нерозривний дефіс в назві «КПІ-2021».

Нерозривний пробіл (*Nonbreaking space*) вставляється комбінацією клавіш *Ctrl+Shift+пробіл*.

Нерозривний дефіс (*Nonbreaking hyphen*) вставляється комбінацією клавіш *Ctrl+Shift+дефіс*.

Позначка трьох крапок вставляється комбінацією клавіш *Ctrl+Alt+крапка*.

Переведення рядка без розривання абзацу проводиться комбінацією клавіш *Shift+Enter*.

Для позначення цитат, власних назв, термінів тощо використовуються лапки.

Редактор має кілька різновидів лапок. Для розкладки клавіатури «рус/укр» клавіши «*Shift+2*» виводять лапки-ялинки «*»*», клавіша «*`*» виводить марровські лапки «*’*». Для розкладки клавіатури «eng» клавіша «*’*» виводить марровські лапки «*’*», клавіши «*Shift+’*» виводять подвійні верхні лапки «*“”*».

Натискання *Ctrl+Z* безпосередньо після друку лапок переводить лапки в прямі.

Введення лапок, як і будь-яких символів, можна провести комбінаціями клавіш *Alt+код*, *код+Alt+X*.

Український правопис рекомендує використовувати лапки-ялинки. У випадку цитати всередині іншої цитати рекомендується вживання лапок-ялинок як зовнішніх лапок, а “лапок-лапок”, англійських – як внутрішніх. У рукописних текстах замість англійських лапок застосовуються німецьки „...“ або особливі, угорські „...” [3].

Вставляння інформації (*Paste*) з буфера обміну (*Clipboard*) може проводитися з додаванням форматування з буферу до форматів документа *Keep source formatting*, з встановленням в новому документі формату за замовчанням *Обычный* та з додаванням форматування з буферу до форматів існуючого документа *Merge formatting*, надання інформації з буферу формату

*Обычный Keep text only.* За замовчанням вставляння з буферу клавішами *Ctrl+V* проводиться з додаванням форматування.

Провести вставляння можна спочатку обравши спосіб вставляння в одному з пунктів **Home-Paste**, а потім вставити фрагмент.

Можна спочатку вставити фрагмент з буферу. Потім натиснути клавішу *Ctrl*. На місці вставленого фрагменту з'явиться меню. Обрати спосіб вставляння можна кнопками меню або клавішами: «В» – зберегти вихідне форматування, «J» – поєднання форматів, «C» – тільки текст.




Формат абзаців (*Paragraph styles*) визначає шрифт символів, порядок вирівнювання слів в рядку (*Alignment*), режим проставляння переносів (*Hyphenation*), горизонтальне положення абзацу на сторінці (*Indentation*), міжрядковий інтервал (*Line spacing*), інтервали між абзацами (*Spacing*) тощо.



Вирівнювання слів в рядку визначається кнопками **Home-Paragraph** стрічки або полем **General-Alignment** вікна налаштувань абзаців **Paragraph**. Вікно активується пунктом контекстного меню абзацу або *Dialog Box launcher* стрічки.

Рекомендованим є вирівнювання по ширині рядка *По ширине (Justify)*.

Режим проставляння знаку перенесення слів між рядками може бути встановлений як ручний користувачем *Manual*, автоматичний *Automatic*, або без перенесення *None*. Режим встановлюється полями **Layout-Hyphenation** стрічки. Режим автоматичного розташування слів в рядку абзацу називається *Word Wrap*.

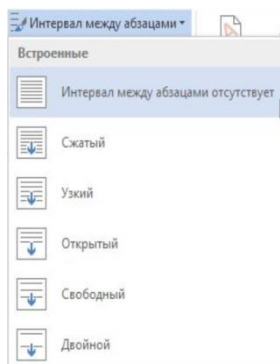
Рекомендованим є режим без перенесення *None*.

Горизонтальне розташування рядків абзаців помічається на горизонтальній лінійці (рис. 1.1). Відступ першого рядка (*first-line indent*) від лівого краю абзацу помічається маркером . Відступ краю абзацу від лівого берега (*left indent*) позначається маркером . Відступ краю абзацу від правого берега (*right indent*) позначається маркером .

Змінити значення горизонтальних відступів можна безпосереднім зсувом маркерів. Зсув маркера  за нижній край *Left intend* переміщує весь абзац. Зсув маркера  за верхній край *Hanging indent* переміщує абзац без зміни положення маркера відступу першого рядка.

Редактор передбачає автоматичне уникнення такого розташування абзацу на сторінці, коли на ній знаходиться тільки один рядок абзацу. Режим контролю першого та останнього рядків (*Widow, orphan*) на сторінці вмикається на сторінці **Line and Page Breaks-Widow/Orphan control** вікна налаштування абзаців.

Можна використовувати визначені наперед формати абзаців [2]:



між абзацами 0pt, міжрядкова відстань 1 *No.*  
 перед 0pt, після 4pt, міжрядкова відстань 1 *Compact*  
 перед 0pt, після 6pt, міжрядкова відстань 1.15 *Tight*  
 перед 0pt, після 10pt, міжрядкова відстань 1.15 *Open*  
 перед 0pt, після 6pt, міжрядкова відстань 1.5 *Relaxed*  
 перед 0pt, після 8pt, міжрядкова відстань 2 *Double*.

Вибір проводиться пунктом **Design-Paragraph Spacing** стрічки.

Визначати параметри тексту можна безпосередньо під час введення. При цьому будь-яка зміна формату автоматично відображається в списку форматів документа. Такий спосіб створення документа не можна вважати ефективним з причини збільшення витраченого часу, збільшення розміру самого документа, більшої складності редагування в подальшому.

Рекомендується попередньо визначити стилі абзаців, які передбачається використовувати в документі. Зазвичай таких стилів небагато. Визначені стилі розміщуються в палітрі стилів (*Styles gallery*) на головній закладці стрічки. Створення документа за такої технології відбувається встановленням курсору в місце введення тексту, обранням стилю абзацу чи тексту щигликом на відповідному стилі з бібліотеки та наступним введенням тексту. Зміна стилю вже введеного тексту проводиться простим щигликом на відповідному стилі з бібліотеки.

### Горизонтальні лінії

При оформленні документів застосовуються *горизонтальні лінії*, які відокремлюють частини документа.

Провести горизонтальну лінію можна автоформатуванням, визначенням властивостей абзацу, безпосереднім малюванням лінії як фігури та створенням таблиці.

Рекомендованим виробником є спосіб малювання горизонтальної лінії шляхом автоформатування. Лінія зображується на всю ширину рядка. Зміна розмірів такої лінії є неможливою. Зображується лінія після трикратного введення символу та натискання клавіші *Enter*:

символи «*» генерують пунктирну лінію,	.....
символи «=» генерують подвійну тонку лінію,	=====
символи «~» генерують хвилясту лінію,	~~~~~
символи «#» генерують подвійну товсту лінію,	=====
символи «-» генерують товсту лінію,	=====
символи «_» генерують тонку лінію.	=====

Прибрати лінії відразу можна гарячими клавішами відміни дії *Ctrl+Z*.

Прибрати лінію можна встановивши курсор на рядок над лінією та обравши пункт **Main-Paragraph group menu Borders- No border** стрічки.

Автоформатування символів для малювання горизонтальних ліній вмикається пунктом **Options-Proofing-Autocorrect options-Autoformat-Apply as you type-Border lines** стрічки.

Абзац документу може бути підкреслений горизонтальною лінією або взятий в рамку.

Лінія підкреслення малюється як суцільна пунктом **Main-Paragraph group menu Borders- Horizontal line**. Лінія є графічним об'єктом, який можна пересунути в інше місце, змінити його розміри, видалити. Контекстне меню лінії дозволяє налаштувати довжину, ширину, колір, стиль позиціонування лінії.

Рамка навколо абзацу малюється визначеним візерунком. Лінії рамки можуть бути намальовані навколо будь яких боків абзацу. Ширина, колір, візерунок рамки абзацу визначається на сторінці рамок контекстного меню,

яке викликається пунктом стрічки **Main-Paragraph group menu Borders-Borders and Shading**.

Будь який фрагмент тексту може мати рамку. Рамка навколо тексту задається пунктом стрічки, як і для абзацу. Рамкою можна тільки охопити текст з усіх боків.

Горизонтальна лінія є різновидом графічного елементу фігури (*Shape*). Вставлення фігури проводиться пунктом **Insert-Shapes-Lines** стрічки (рис. 1.4). Визначення розмірів, положення на сторінці, кольору, візерунку фігури тощо визначається на сторінці форматування **Format**. Сторінка автоматично активується при виділенні фігури.

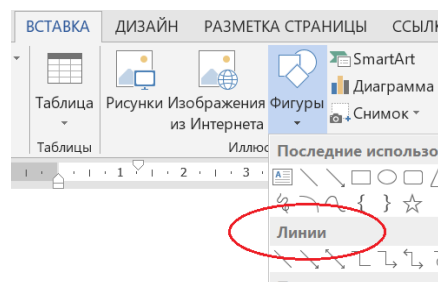


Рис. 1.4. Фігури

### 1.1.2. Завдання для самостійного виконання

#### Завдання 1.1.1. Табуляція.

Набрати текст інформаційної довідки (рис. 1.5). Орієнтація аркуша альбомна Шрифт *Calibri* 11pt з визначенням позицій табуляцією. Файл зберегти як текстовий.

Рядок «План»: вирівнювання по центру, перша позиція 5.75 см, крок 2 см.

Інформаційні рядки факультетів: стовпці зі значеннями одного порядку табулювати по лівому краю, стовпці зі значеннями різних порядків табулювати по правому краю.



Результати вступних кампаній 2015 – 2018 років  
 Національний Технічний Університет України «Київський політехнічний інститут імені Ігоря Сікорського»  
 спеціальність 151 «Автоматизація та комп'ютерно інтегровані технології»

	2015			2016			2017			2018		
	план	заяв	факт	план	заяв	факт	план	заяв	факт	план	заяв	факт
Теплотехнічний факультет	40	406	45	24	259	22	52	466	50	57	457	57
Хіміко-технічний факультет	25	605	25	56	384	46	28	143	18	28	102	13
Приладобудівний факультет	74	330	76	80	592	80	100	477	100	110	617	106
Інженерно-хімічний факультет	48	205	48	60	259	60	52	466	50	57	211	44

Рис. 1.5. Інформаційна довідка до завдання 1.1.1

*Завдання 1.1.2. Форматування.*

Набрати наказ (рис. 1.6) з наступними параметрами документа:

- береги згори та знизу 2 см, справа 1.5 см, зліва 3 см;
- міжрядковий інтервал 1.15;
- зображення герба: висота 2 см, розташування по центру;
- шапка наказу: шрифт Times New Roman Bold 13pt по центру;
- рядок «наказ №»: шрифт Times New Roman Bold 18pt по центру;
- текст наказу: шрифт Times New Roman 12pt по ширині;
- виділений текст : шрифт Times New Roman Bold 12pt по лівому краю.

*Завдання 1.1.3. Стилi.*

Набрати текст титульного аркуша диплому (рис. 1.7) з використанням стилів абзаців. Стилi додати в колекцію експрес-стилів. Береги аркуша 1.5 см.

Стиль «Назва»: шрифт Times New Roman Bold 12pt по центру без відступу першого рядка, міжрядковий інтервал 1, оточуючі відстані 0 пт, всі літери прописані.

Стиль «Назва 1»: шрифт Times New Roman Bold 12pt по центру без відступу першого рядка, міжрядковий інтервал 1, оточуючі відстані 0 пт.

Стиль «Допущено»: шрифт Times New Roman 13pt по лівому краю без відступу першого рядка, міжрядковий інтервал 1, відстані 0 пт, відступ абзацу 10 см.

Стиль «Диплом»: шрифт Times New Roman Bold 20pt по центру без відступу першого рядка, міжрядковий інтервал 1, оточуючі відстані 0 пт, літери прописані.

Стиль «Виконав»: шрифт Times New Roman 13pt по лівому краю без відступу першого рядка, міжрядковий інтервал 1, відстань перед 12 пт.

Стиль «Текст»: шрифт Times New Roman 13pt по лівому краю без відступу першого рядка, міжрядковий інтервал 1, відстань перед 6 пт.

Стиль «Надрядок»: шрифт Times New Roman 13pt по лівому краю без відступу першого рядка, надрядковий, міжрядковий інтервал 1, оточуючі відстані 0 пт.

Стиль «Плагіат»: шрифт Times New Roman 13pt по лівому краю без відступу першого рядка, міжрядковий інтервал 1, оточуючі відстані 0 пт. Відступ абзацу 8 см.

Стиль «Київ»: шрифт - Times New Roman 13pt по центру без відступу першого рядка, міжрядковий інтервал 1, відстань перед 12 пт.



УКРАЇНА  
МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ  
імені ІГОРЯ СІКОРСЬКОГО»

НАКАЗ № \_\_\_\_\_

м. Київ

«\_\_» \_\_\_\_\_ 20\_\_ р.

---

Про затвердження тем і керівників дипломних проектів (дипломних робіт, магістерських дисертацій) здобувачів ступеня \_\_\_\_\_

(бакалавра/магістра)

у 20\_\_ / 20\_\_ навчальному році

---

З метою якісної підготовки до випускної атестації студентів університету

\_\_\_\_\_ рівня вищої освіти

(першого (бакалаврського), другого (магістерського))

НАКАЗУЮ:

1. Затвердити теми та призначити керівників дипломних проектів (дипломних робіт, магістерських дисертацій) студентів університету за поданням факультетів та інститутів (додатки №№ \_\_\_\_\_).

2. Директорам інститутів, деканам факультетів забезпечити систематичний контроль за організацією випускної атестації на підпорядкованих випускових кафедрах.

3. Завідувачам випускових кафедр створити умови для якісної підготовки студентів до випускної атестації.

4. Контроль за виконанням наказу покласти на \_\_\_\_\_  
(посада, прізвище та ініціали відповідальної особи)

Ректор

М. З. Згуровський

Рис. 1.6. Бланк наказу до завдання 1.1.2

**НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ  
ІМЕНІ ІГОРЯ СІКОРСЬКОГО»**

\_\_\_\_\_ (повна назва інституту/факультету)

\_\_\_\_\_ (повна назва кафедри)

«До захисту допущено»

Завідувач кафедри

\_\_\_\_\_ (підпис)

\_\_\_\_\_ (ініціали, прізвище)

“ \_\_\_ ” \_\_\_\_\_ 20\_\_ р.

**Дипломний проєкт  
на здобуття ступеня бакалавра**

спеціальності \_\_\_\_\_ (код та назва спеціальності)

на тему: \_\_\_\_\_

Виконав (-ла): студент (-ка) \_\_\_\_\_ курсу, групи \_\_\_\_\_ (шифр групи)

\_\_\_\_\_ (прізвище, ім'я, по батькові)

\_\_\_\_\_ (підпис)

Керівник \_\_\_\_\_ (посада, науковий ступінь, вчене звання, прізвище та ініціали) \_\_\_\_\_ (підпис)

Консультант \_\_\_\_\_ (назва розділу) \_\_\_\_\_ (посада, вчене звання, науковий ступінь, прізвище, ініціали) \_\_\_\_\_ (підпис)

Рецензент \_\_\_\_\_ (посада, науковий ступінь, вчене звання, науковий ступінь, прізвище та ініціали) \_\_\_\_\_ (підпис)

Засвідчую, що у цьому дипломному проєкті немає запозичень з праць інших авторів без відповідних посилань.

Студент \_\_\_\_\_ (підпис)

Київ – 20\_\_ року

Рис. 1.7. Титульний аркуш диплому до завдання 1.1.3

## Практикум 1.2. Оформлення таблиць та зображень в MS Word

### 1.2.1. Теоретичні положення

Таблиці є способом організації та візуалізації інформації, яка може бути структурована по рядках та стовпцях.

Компоненти таблиці: стовпці, рядки, окремі комірки, - можуть мати надані власні формати, рамки, кольори та інші елементи декору.

Операції роботи з таблицями є зрозумілими та інтуїтивними. Створення таблиць проводиться з меню стрічки **Insert-Tables-Add a Table**. Форматування, редагування таблиці проводиться пунктами контекстного меню таблиці або з закладок **Design** та **Layout** стрічки, які автоматично активуються при виділенні таблиці.

Таблиці *MS Word* є не тільки способом групування даних, але й засобом проведення розрахунків. Над значеннями в комірках таблиці можна проводити математичні операції.

Арифметичні вирази в таблиці вводяться у вікні формул (рис. 1.8 а). Вікно активується пунктом **Layout-Data-formula** стрічки (рис. 1.8 б). Адреси комірок таблиці вводяться за правилами *MS Excel*.

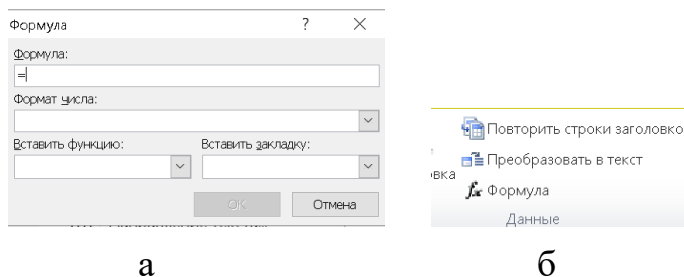


Рис. 1.8. Керування таблицями: а – вікно формул; б – виклик вікна формул

Автоматично розраховується значення в комірці тільки після введення формули. Для подальших перерахунків слід виділяти потрібні комірки та натискати клавішу *F9*.

Виділити всю таблицю можна кліком на позначці «хрест» таблиці. 

## 1.2.2. Завдання для самостійного виконання

### Завдання 1.2.1. Таблиці.

а) Набрати таблицю розкладу (рис. 1.9) з дотриманням стилів та кольорів.

б) Набрати на одному аркуші засобами колонок два товарні чеки (рис. 1.10). Для «Місця печатки» використати графічну форму.

### Завдання 1.2.2. Розрахункові таблиці.

Набрати таблицю результатів вступу (рис. 1.11). Рядок «КПІ», стовпці «конкурс», «к», «%» зробити РОЗРАХУНКОВИМИ, такими, що ОБЧИСЛЮЮТЬСЯ РЕДАКТОРОМ. Коефіцієнт рахувати як  $k = \frac{\text{заяв.}}{\text{план}}$ . Відсоток рахувати як відношення факту до плану. Вивести в документ дві діаграми.

	Понеділок	Вівторок	Середа	Четвер	П'ятниця
08:30	<b>Матеріалознавство</b> Заєць С. С. 1-201-01	<b>Історія України</b> Перга Ю. М. 1-217-01		<b>Інженерна графіка</b> Баскова Г. В. 835-07	
10:25	<b>Іноземна мова</b> Іванова О. О. 1-205-01	<b>Програмування</b> Павловський О. М. 1-201-01	<b>Фізика</b> Родіонов В. М. 1-215-01	<b>Вища математика</b> Суліма О. В. 1-201-01	<b>Інформаційні технології</b> Боровицький В. М. 1-201-01 <b>Історія України</b> Рамазанов Ш. Ш. 1-201-01
12:20	<b>Інформаційні технології</b> Кравченко І. В. 32-13	<b>Вища математика</b> Суліма О. В. 1-217-01	<b>Фізика</b> Генкін О. М. 1-201-01	<b>Вища математика</b> Суліма О. В. 1-203-01	<b>Програмування</b> Лакоза С. Л. 1-291-01 <b>Інженерна графіка</b> Баскова Г. В. 105-07
14:15	<b>Матеріалознавство</b> Заєць С. С. 172-01	<b>ФП</b>			<b>Програмування</b> Лакоза С. Л. 1-291-01

Рис. 1.9. Таблиця до завдання 1.2.1 а

**ТОВАРНИЙ ЧЕК**

Продавець \_\_\_\_\_  
 ІНП \_\_\_\_\_  
 Кор. рахунок \_\_\_\_\_ МФО \_\_\_\_\_  
 Р.Р. \_\_\_\_\_  
 Тел./ Факс \_\_\_\_\_

№	Найменування	Ціна	Кількість	Сума

Податковий Рахунок-фактура № \_\_\_\_\_ від \_\_\_\_\_ 2018р.  
 Покупець \_\_\_\_\_ ІНП \_\_\_\_\_  
 Пункт призначення \_\_\_\_\_ довіреність \_\_\_\_\_

№	Найменування робіт та послуг	Вартість без ПДВ	ПДВ		Всього
			Ставка	Сума	
1	Найменування товарів, які оподатковуються ПДВ				
2	Найменування товарів, які не оподатковуються ПДВ				
3	Найменування послуг, які оподатковуються ПДВ				
4	Найменування товарів, які оподатковуються ПДВ				
5	<b>Всього по оподаткованому обсягу</b>				
6	<b>Всього по неоподаткованому обсягу</b>				

П.І.Б. виконавця \_\_\_\_\_ Місце печатки \_\_\_\_\_ Підпис \_\_\_\_\_

**ТОВАРНИЙ ЧЕК**

Продавець \_\_\_\_\_  
 ІНП \_\_\_\_\_  
 Кор. рахунок \_\_\_\_\_ МФО \_\_\_\_\_  
 Р.Р. \_\_\_\_\_  
 Тел./ Факс \_\_\_\_\_

№	Найменування	Ціна	Кількість	Сума

Податковий Рахунок-фактура № \_\_\_\_\_ від \_\_\_\_\_ 2018р.  
 Покупець \_\_\_\_\_ ІНП \_\_\_\_\_  
 Пункт призначення \_\_\_\_\_ довіреність \_\_\_\_\_

№	Найменування робіт та послуг	Вартість без ПДВ	ПДВ		Всього
			Ставка	Сума	
1	Найменування товарів, які оподатковуються ПДВ				
2	Найменування товарів, які не оподатковуються ПДВ				
3	Найменування послуг, які оподатковуються ПДВ				
4	Найменування товарів, які оподатковуються ПДВ				
5	<b>Всього по оподаткованому обсягу</b>				
6	<b>Всього по неоподаткованому обсягу</b>				

П.І.Б. виконавця \_\_\_\_\_ Місце печатки \_\_\_\_\_ Підпис \_\_\_\_\_

Рис. 1.10. Чек до завдання 1.2.1 б

	2015					2016					2017					2018				
	план	заяв	факт	k	%	план	заяв	факт	k	%	план	заяв	факт	k	%	план	заяв	факт	k	%
Теплотехнічний факультет	40	406	45	10,2	113%	24	259	22	10,8	92%	52	466	50	9,0	96%	57	457	57	8,0	100%
Хіміко-технічний факультет	25	605	25	24,2	100%	56	384	46	6,9	82%	28	143	18	5,1	64%	28	102	13	3,6	46%
Приладобудівний факультет	74	330	76	4,5	103%	80	592	80	7,4	100%	100	477	100	4,8	100%	110	617	106	5,6	96%
Інженерно-хімічний факультет	48	205	48	4,3	100%	60	259	60	4,3	100%	52	466	50	9,0	96%	57	211	44	3,7	77%
<b>КПІ</b>	<b>187</b>	<b>1546</b>	<b>194</b>	<b>10,8</b>	<b>104%</b>	<b>220</b>	<b>1494</b>	<b>208</b>	<b>7,4</b>	<b>95%</b>	<b>232</b>	<b>1552</b>	<b>218</b>	<b>7,0</b>	<b>94%</b>	<b>252</b>	<b>1387</b>	<b>220</b>	<b>5,5</b>	<b>87%</b>

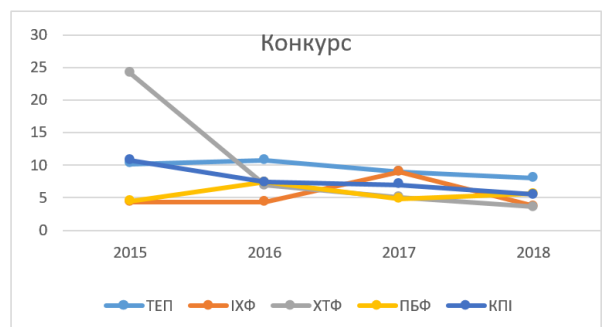


Рис. 1.11. Завдання 1.2.2

## Практикум 1.3. Оформлення науково-технічних матеріалів в MS Word

### 1.3.1. Теоретичні положення

Колонтитули (*Header and Footer*) – горизонтальні смуги згори та знизу сторінки. Інформація, яка розміщується в колонтитулах, повторюється на всіх сторінках розділу документа або всього документа. В колонтитулах зазвичай розміщуються номери сторінок, назви розділів тощо. Саме за

рахунок розбиття документа на розділи та застосування колонтитулів можна оформити аркуш у відповідності до вимог Системи конструкторської документації [4].

Для вставлення рамки аркуша та основного напису за ДСТУ ГОСТ 2.104:2006 можна скористатися кількома способами. Перша можливість полягає в застосуванні малюнку рамки в якості *фону* (*подложка, background*). Другий спосіб – вставлення малюнку рамки в нижній колонтитул аркуша. Третій спосіб найбільш трудомісткий – вставлення в колонтитул рамки у вигляді таблиці.

Нумерація сторінок внизу або нагорі сторінки найпростіше проводиться кнопкою стрічки **Insert-Header&Footer-Page number**. Кнопка меню стрічки також знаходиться з лівого краю закладки колонтитулів, яка активується автоматично разом з стрічками колонтитулів подвійним кліком згори або внизу сторінки.

Для того, щоб встановити номер сторінки в потрібному місці сторінки слід встановити в це місце курсор та обрати пункт меню стрічки **Insert-Header&Footer-Page number-Current position**.

Номер сторінки є одним з вбудованих *елементів-полів* (*field*) редактора з назвою *PAGE*: **Insert-Quick parts-Field- PAGE**.

Застосування замість Експрес блока фігури, в яку вставлене поле *PAGE* дозволить визначити місце точніше: **Insert-Shapes- Main shapes-Text box- Field- PAGE**.

## Зміст

Зміст (*Table of contents, Оглавление*) відноситься до розділів документа. Розміщується зміст на першому аркуші документа. Умовою для автоматичного створення є наявність в структурних частинах документу стилів *ЗаголовокXXX*. Створюється зміст відповідним полем **References-Table of contents** стрічки.

## Цитування

Керування цитуванням проводиться напівавтоматично через поле **Citations&Bibliography** стрічки (Рис. 1.12).



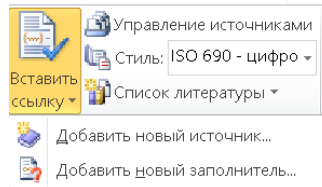


Рис. 1.12. Керування цитуваннями

Редактор накопичує бібліографічні дані джерел у власній базі. Введені один раз джерела зберігаються та можуть бути використані в інших документах.

Додавання нових джерел проводиться пунктом **References-Citation&Bibliography-Add new Source** (рис. 1.12).

Бібліографічні дані джерела вводяться в картку джерела.

Редактор має змогу проставляти напівавтоматично внутрішньотекстові та позатекстові посилання.

Проставлення цитування в текст проводиться обранням зі списку джерел бази в полі **References-Citation&Bibliography-Insert Citation** (рис. 1.13).

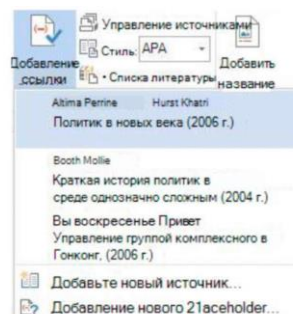


Рис. 1.13. Список джерел для цитування

Кількість інформації в цитування можна редагувати обравши пункт **Edit Citation** контекстного меню посилання.

За умови цитування вбудованими засобами можливо напівавтоматично створити список використаної літератури. Для цього треба помістити курсор в потрібне місце документа, обрати стиль цитування та пункт **References-Bibliography** стрічки.

Бібліографію можна скорегувати після додавання нового цитування у вже форматований документ. Для цього треба клацнути на бібліографії та

обрати пункт **Update Citations and Bibliography** контекстного меню.

Вбудований пункт оформлення *ГОСТ ...* НЕ ВІДПОВІДАЄ вимогам. Найближче до вимог ДСТУ 3008:2015 [5] є пункт *стиль IEEE*.

Засоби підрядкового цитування реалізовані у вигляді *виносок (footnote)*. Керування виносками проводиться з групи **References-Footnotes** стрічки.

Виноски НЕ ВРАХОВУЮТЬСЯ АВТОМАТИЧНО В БАЗУ ДЖЕРЕЛ та СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.

Формат виносок редактора не відповідає вимогам ДСТУ 3008:2015 по розміру шрифту, міжрядковому інтервалу, позначці виноска та потребує коригування.

### Складені документи

Орієнтуватися в документі, обсяг якого сягає кількох сотень сторінок, стає важко. Для роботи з такими документами пропонується застосовувати технологію складених документів (*Complex document*) та режим структури.

Складений документ – документ, який зберігається у вигляді кількох файлів. Розбивається складений документ за розділами. Зберігається складний документ у вигляді головного файлу та файлів, кожен з яких містить один розділ. Такі файли мають назву розділів. Зображується складений документ в редакторі у вигляді гіперпосилань на файли розділів.

В документі в режимі структури абзаци позначаються сірим колом на початку першого рядка зліва. Рядки заголовків частин всіх рівнів (розділи, підрозділи, параграфи тощо) помічаються маркером дії **+**.

Роботою зі складеним документом керує закладка **Outline** (рис. 1.14). Стрічка активується полем **View-Outline** стрічки.

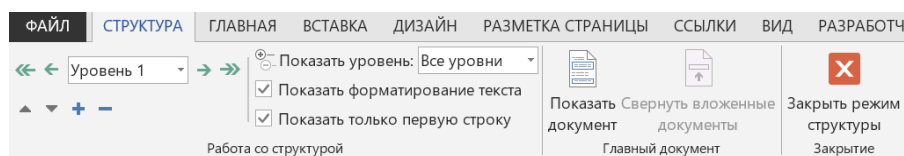


Рис. 1.14. Стрічка структури

Поле **Show first line only** стрічки можна залишати на екрані тільки перший рядок абзаців. Обраний елемент списку **Show level** визначає максимальний рівень розділів, заголовки яких будуть виведені на екран. Елемент списку **All levels** виводить на екран текст всього документа.

Кліком на маркері дії або на кнопках «+» та «-» стрічки можна відкрити або сховати текст виділеного розділу.

Кнопками у вигляді стрілок стрічки можна пересувати обраний розділ вгору та вниз по дереву структури документа.

Поле **Level** стрічки показує рівень заголовка розділу. Стрілками злів та справа від поля можна підвищити або знизити рівень розділу.

Кнопка **Show Document** активує інші кнопки групи **Master Document**: створити **Create** та вставити **Insert**.

З виділеного розділу кнопкою **Create** створюється зовнішній файл з назвою розділу.

Кнопка **Insert** вставляє зовнішній файл формату *.docx* в структуру документа.

### **1.3.2. Завдання для виконання**

#### *Завдання 1.3.1. Оформлення аркуша.*

Ввести в документ основний напис (рис. 1.15) за ДСТУ 2.104:2006 та проставити в ньому авторство, номер сторінки: а) фоном; б) рисунком; в) таблицею.

#### *Завдання 1.3.2. Оформлення науково-технічної публікації.*

Оформити наукову публікацію за ДСТУ 3008:2015. За основу можна взяти будь-яку опубліковану в науково-технічному журналі статтю. Публікація має містити не менше трьох абзаців тексту, двох формул з наведених на рис. 1.16, однієї схеми з наведених на рис. 1.17, не менше чотирьох цитувань, оформлених за ДСТУ 8302:2015 [6], список використаних джерел. Посилання оформити як одне внутрішньотекстове

одне підрядкове, два позатекстових. Джерела посилань можна дублювати. Список джерел, посилання, схеми, формули сформувати засобами редактора.

Параметри документа:

Верхній та нижній береги 25 мм, лівий – 18 мм, правий – 22 мм. Гарнітура шрифту Times New Roman. У верхньому лівому куті аркуша необхідно вказати індекс УДК (шрифт розміром 12 пт), нижче посередині вказати назву статті (шрифт розміром 14 пт всі літери прописані, інтервал – одинарний), далі через інтервал посередині вказати прізвище автора, організацію, місто, країну (шрифт розміром 12 пт, курсив), нижче через інтервал – текст (шрифт розміром 14 пт, інтервал – одинарний), наприкінці вказати ключові слова. Формули та позначення набирати за допомогою редактора формул як окремий об'єкт розмірами: звичайний – 14 пт, великий індекс – 7 пт, малий індекс – 5 пт, великий символ – 18 пт, малий символ – 12 пт, кириличні та грецькі літери – прямі.

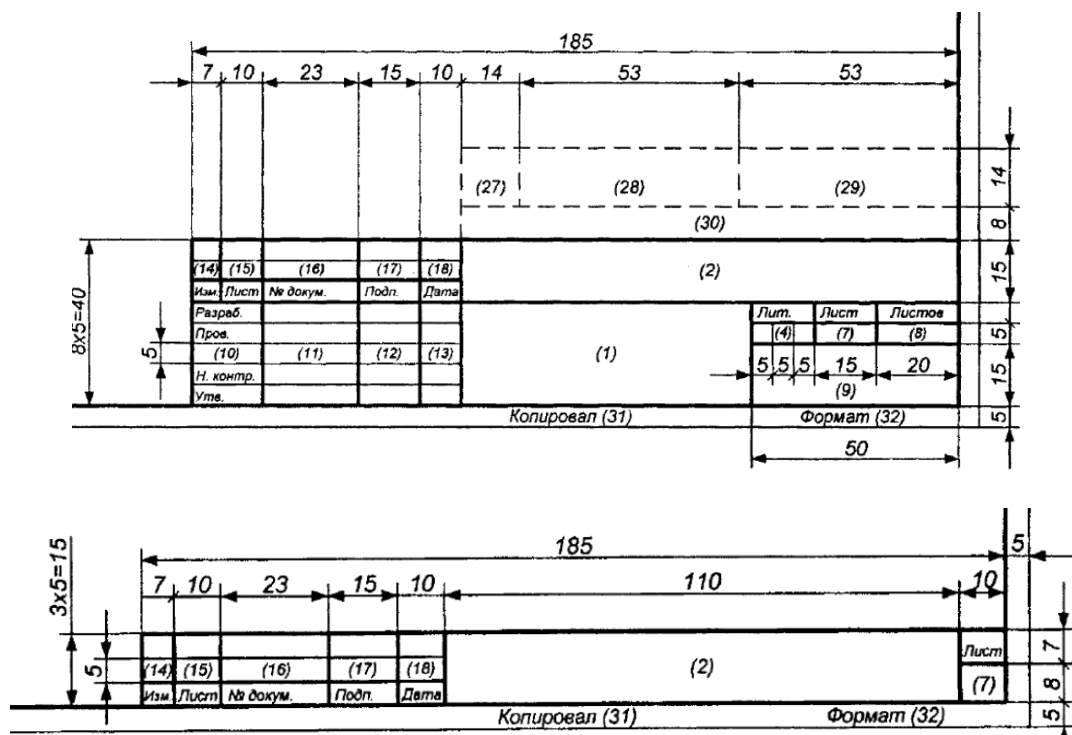


Рис. 1.15. Основний напис за ДСТУ 2004:2006 до завдання 1.3.1 [4]

Завдання 1.3.3. Створити з файлу complex.docx складений документ, в якому розділи розміщуються в окремих файлах. Згенерувати зміст. На горі сторінки згенерувати назву розділу.

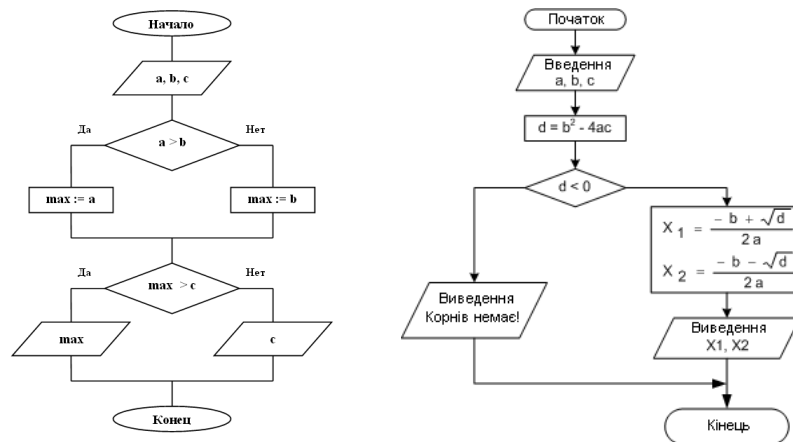


Рис. 1.16 – Схеми до завдання 1.3.2

$$LSF_v(x) = \begin{cases} \frac{1}{\pi\sqrt{X_{0v}^2 - x^2}}, & \text{коли } |x| < X_{0v} \\ 0, & \text{коли } |x| \geq X_{0v} \end{cases}$$

$$p = \sum_{k=l+1}^{\infty} \frac{(k\tau)^k}{k!} e^{-v\tau} = 1 - \sum_{k=0}^{\infty} \frac{(k\tau)^k}{k!} e^{-v\tau}$$

$$NETD = \frac{4k_{eff}^2}{\tau_0 \int_{\lambda_1}^{\lambda_2} D^*(\lambda) \frac{\partial M_\lambda(\lambda, T)}{\partial T} d\lambda} \sqrt{\frac{\Delta f}{A_D}}$$

$$P_{U_{eb}}(k, x, y) = \frac{\exp\left(-\frac{U_{eb}(x, y)}{2\sigma_{eb}}\right)}{2\Gamma\left(\frac{k}{2}\right)} \left(\frac{U_{eb}(x, y)}{2\sigma_{eb}}\right)^{\frac{k}{2}-1}$$

Рис. 1.17 – Формульні вирази до завдання 1.3.2

## Практикум 2.1. Форматування таблиць в MS Excel

### 2.1.1. Теоретичні положення

#### Робоче вікно процесора

В робочому вікні процесора (рис. 2.1) розміщуються органи керування та робоче поле. Керування редактором проводиться з панелі швидкого доступу *Quick Access Toolbar*, стрічки *Ribbon*, поля назв *Name Box*, рядка введення *Formula bar*, рядка стану *Status bar*. На стрічці елементи керування згруповані на закладках *Tabs* відповідно до виконуваних дій [7].

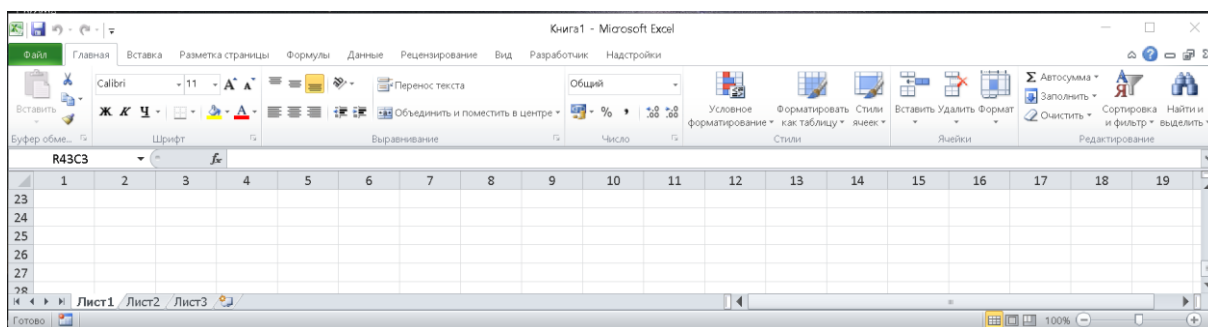


Рис. 2.1. Робоче вікно редактора

Згорнути/розгорнути стрічку можна клавішами *CTRL+F1*, або пунктами кнопки **Ribbon view control** та контекстного меню стрічки.

Більшість елементів керування та виділені в документі фрагменти мають власні контекстне меню (*Shortcut menu*). Склад меню змінюється в залежності від дій, які можна виконати з обраним елементом. Відкривається контекстне меню правим щикликом миші над обраним елементом.

Рядок стану знаходиться внизу робочого вікна. В ньому містяться елементи керування зображенням документу. Типовою конфігурацією рядку стану є кнопки зображення документу (*View controls*) та кнопки керування масштабом зображення документу (*Zoom controls*). Склад елементів рядка статусу можна змінювати в контекстному меню рядка.

На панелі швидкого доступу розміщуються кнопки команд, які виконуються найчастіше. Зазвичай це є команди збереження документу, відміни останніх дій, повторення останньої дії. Користувач має змогу змінити склад кнопок на панелі швидкого доступу. Сюди зручно додавати кнопки запуску макросів користувача. Режим редагування панелі швидкого

доступу вмикається пунктом **Customize Quick Access Toolbar**, контекстним меню стрічки або пунктом **File-Options-Quick Access Toolbar**.

Документи *MS Excel* називаються робочими книгами (*workbook*). Документи зберігаються в файлах (*spreadsheet file*) з розширенням *.xls*, *.xlsx*, *.xlsm*, *.xlt*, *.xltn*, *.ltx*, *xlsb*. Кожен документ може містити кілька аркушів (*worksheet*) з таблицями, діаграмами, макросами тощо. Сторінки документа помічаються внизу робочого поля закладками. За замовчанням сторінки мають імена *Лист1*, *Лист2* тощо (*Sheet1*, *Sheet2*). Користувач може надати свої імена сторінкам через відповідним пункт контекстного меню сторінки.

На робочому полі редактора зображується сторінка документу. Документ може бути зображений в звичайному режимі *Normal*, режимі розмітки сторінки *Page Layout*, сторінковий *Page Impression*.

В полі назв (*Name Box*) відображається адреса поточної комірки та задаються власні імена комірок.

В рядку введення (*Formula bar*) відображається зміст поточної комірки та вводяться формульні вирази для проведення обчислень в активній комірці.

### **Адресація комірок**

Електронну таблицю створюють рядки (*row*) та стовпці (*column*), які мають свої імена – адреси. Перетин рядка та стовпця показує комірку таблиці (*cell*).

Для вказання на групу суміжних комірок використовують блок комірок.

В *MS Excel* використовується два стилі адрес: «традиційну» «A1» та «програмну» «R1C1».

В «традиційному» стилі спочатку вказується ім'я стовпця, потім – рядка. Імена рядків визначаються їхніми номерами. Нумерація починається з 1. Імена стовпців визначаються літерами латинського алфавіту спочатку від A до Z, потім від AA до AZ, потім від BA до BZ тощо.

Ідентифікація комірок проводиться їхньою адресою перетину відповідних рядка та стовпця, наприклад, **C4**.

Адреса блоку визначається адресами комірки в лівому верхньому куті блоку та комірки в нижньому правому куті блоку. Між адресами ставиться знак двокрапки «:». Наприклад, **B2:E4**.

В «програмному» стилі спочатку вказується ім'я рядка, потім – стовпця. Імена рядків визначаються літерою R. Імена стовпців визначаються літерою C.

Ідентифікація комірок проводиться їхньою адресою перетину відповідних рядка та стовпця. Вказується номери рядка з префіксом R та стовпця з префіксом C. Нумерація починається з 1. Наприклад, **C4~R4C3**.

Встановлюється стиль адресації полем **File-Options-Formulas-R1C1 reference style** стрічки.

Повна адреса комірки складається з назви аркуша та адреси комірки або групи комірок на аркуші. Наприклад, **Лист1!A1:C3**, **Sheet3!R3C3**.

Поточною, активною (*active cell*) називається комірка, в якій знаходиться курсор. Активна комірка виділяється товстою рамкою. Зміст активної комірки додатково виводиться в рядку введення.

Програмний стиль дозволяє гнучко адресувати комірки відносно поточної чисельними індексами. Число без дужок визначає абсолютний номер рядка/стовпця. Наприклад, **R3C3** визначає комірку в третьому рядку та третьому стовпці. Число в квадратних дужках визначає зсув відносно поточної комірки. Наприклад, **R[1]C[-1]** визначає комірку, яка зсунута на 1 рядок вниз та 1 стовпець вліво. Символ без індексу визначають весь поточний рядок/стовпець. Наприклад, **RC** визначає поточну комірку, **RC[3]** визначає комірку в поточному рядку, яка зсунута на 3 вправо.

## Формули


Формули для розрахунків являють собою сукупність операторів дій та операндів. Операторами слугують символи арифметичних та логічних дій, імена вбудованих функцій та функцій користувача. В якості операндів (аргументів) використовуються числа та адреси комірок, в яких знаходяться дані.



Ознакою проведення обчислень в комірці є наявність символу порівнювання першим елементом в комірці та формульного виразу після знаку порівнювання.

Результат обчислення за формулою розміщується в тій же комірці, в якій розміщено формулу.

Порядок дій в формулах без дужок є традиційним: логічне заперечення, функція, піднесення до степеню, множення та ділення, додавання та віднімання, конкатенація рядків, порівняння (=, <>, <=, >=, <, >). Дії проводяться зліва-направо.

Імена вбудованих функцій можуть вводитися вручну в комірці, в рядку введення або обиратися зі списків вікна вбудованих функцій чи списків відповідного типу функцій групи **Formulas-Function Library** стрічки. Вікно вбудованих функцій активується кнопкою  зліва від рядка введення або кнопкою **Formulas-Insert Function** стрічки.

Імена функцій є локалізованими. Тобто вводяться мовою, яка співпадає з мовою локалізації *MS Excel*.

За замовчанням формули розраховуються при їхньому введенні та автоматично перераховуються при зміні будь-якого з операндів формули. Режим обчислення визначається полями **Formulas-Calculation-Calculation Options** стрічки.

При застосуванні формул слід мати на увазі, що незалежно від стилю адресації в формулах використовуються *відносні* від поточної комірки посилання.

При копіюванні комірок в нове місце в формулах автоматично перераховуються адреси операндів таким чином, що ЗСУВИ ОПЕРАНДІВ НЕ ЗМІНЮЮТЬСЯ.

Для управління поведінкою адрес операндів в формулах введені абсолютні (*absolute reference*), абсолютні по рядку або стовпцю та відносні (*relative reference*) адреси.

Абсолютна адреса – це незмінна при копіюванні адреса. Позначається абсолютна адреса символом «\$». Наприклад, **\$A\$1** – абсолютне посилання, **A\$1** – абсолютне по рядку 1, **\$A1** – абсолютне по стовпцю А.

Відносна – це адреса, яка змінюється при копіюванні. Наприклад, стандартна адреса A1 є відносною та буде змінюватися при копіюванні.

Швидко змінити тип адреси можна натисканням клавіши *F4* на операнді. Зміна проводиться циклічно. Наприклад, **\$A\$1**→**\$A1**→**A\$1**→**A1**→**\$A\$1**.

Подвійний щиглик на комірці з формулою візуалізує комірки операндів на екрані.

Аргументи функцій можуть визначаються послідовністю адрес комірок через кому, адресою блока комірок, поєднанням блоків комірок через кому, перетином блоків комірок через пробіл.

В комірках можуть міститися символні (текстові), чисельні дані, формули, дати. В комірках з формулами відображається тільки результат, самі формули відображаються за замовчанням в рядку введення в разі виділення комірки.

Вигляд змісту комірок визначається типом даних та форматом відображення, який задано для них. Основними є наступні формати.

*Основний* **General** формат є універсальним для чисел та текстів. Відображення відбувається в тому вигляді, як вводяться дані. Є форматом комірок за замовчанням.

Формат з фіксованою кількістю десяткових знаків **Number**. Використовується для чисельних даних.

*Грошовий* формат **Currency** є різновидом основного. Візуально кожні три розряди числа відокремлюються комою. В кінці чисел проставляється позначка грошової одиниці.


*Відсотковий* формат **Percentage** виводить чисельне значення помноженим на 100 та з позначкою відсотків після значення.


*Науковий* формат **Scientific** є форматом чисел у стандартній формі з плаваючою точкою  $x.xxx \cdot 10^{xx}$ . Застосовується для показу дуже великих або дуже маленьких чисел.

Дані в *текстовому* форматі **Text** відображаються так, як вводяться. Обробляються дані в текстовому форматі як рядки символів.

Визначається формат даних в комірці в діалоговому вікні форматування комірок **Cell Format**. Вікно активується відповідним пунктом контекстного меню комірки, кнопкою запуску діалогового вікна *Dialog Box Launcher* групи стрічки **Home-Number** або списком з тієї ж групи.

Користувач може задати свій власний формат. Формат визначається шаблоном у вигляді послідовності символів діезу для символу, нуля для цифри, коми, пробілу, знаку відсотків тощо.

Швидко змінити кількість знаків в числах в комірках для відображення можна кнопками  групи **Home-Number** стрічки.

Символ десяткового роздільника дійсних чисел змінюється кнопкою  групи **Home-Number** стрічки стрічки.

Зручним інструментом введення послідовностей на кшталт арифметичних прогресій є *режим заповнення*. В режимі заповнення визначення змісту комірок та їхнє форматування проходить автоматично. Вмикається режим заповнення полем **Home-Editing-Fill** стрічки або позначкою заповнення у вигляді хреста в правому нижньому куті виділених комірок.

Для надання формулам більш наочного вигляду та у випадках, коли комірки використовуються багаторазово, слушно замінити їхні адреси іменем.

Для швидкої роботи з іменованими комірками слугує поле назв. В полі назв відображається список імен поіменованих комірок. З цього списку кліком можна обрати ім'я комірки для вставляння в формулу.

Для надання імені комірці достатньо зробити її поточною та в полі назв замість адреси ввести потрібне ім'я.

Повний набір інструментів для роботи з поіменованими коміркам містить група стрічки **Formulas-Defined Names**.

Полем **Define name** можна надати ім'я комірці та визначити область доступності імені, адресу, додати коментар.

Всі можливі дії з поіменованими комірками можна провести у вікні, яке активується полем **Formulas-Defined Names-Names Manager** стрічки.

Щоб тимчасово прибрати рядки або стовці з таблиці можна застосувати дію сховати/показати. Дія виконується для обраних рядків/стовпців відповідним пунктом контекстного меню.

Прибирання елементів з екрану можна провести групуванням. Груповані комірки позначаються на екрані позначкою. Проводиться групування полем **Data-Outline-Group** стрічки.

Засобом захисту від отримання невірних результатів та від аварійного завершення роботи застосунку є перевірка даних при заповненні таблиці. Вмикання режиму перевірки проводиться полем **Data-Data Tools-Data Validation**.

Параметри перевірки даних визначаються у діалоговому вікні, яке має три сторінки. На сторінці *Settings* обирається тип даних, для яких потрібно проводити перевірку, з вбудованого списку обирається логічна умова та задаються відповідні значення.

На сторінці *Input Message* вводиться повідомлення, яке буде виводитися на екран

На сторінці *Error Alert* записується повідомлення, яке буде виводитися на екран в разі помилки введення та визначається вигляд повідомлення.

## **2.1.2. Завдання для самостійного виконання**

### *Завдання 2.1.1. Форматування.*

Створити таблицю результатів вступної кампанії (рис.2.2). Стовпці конкурсу (відношення заяв до плану) та відсотка прийому (відношення факту до плану) зробити розрахунковими. Значення останнього рядка поза рамкою розрахувати як середнє значення відповідного стовпця. Пояснити різницю та визначити правильне значення.

	2015					2016					2017					2018				
	план	заяв	факт	конкурс	%	план	заяв	факт	конкурс	%	план	заяв	факт	конкурс	%	план	заяв	факт	конкурс	%
Теплотехнічний факультет	40	406	45	10.2	113%	24	259	22	10.8	92%	52	466	50	9.0	96%	57	457	57	8.0	100%
Хіміко-технічний факультет	25	605	25	24.2	100%	56	384	20	6.9	36%	28	143	18	5.1	64%	28	102	13	3.6	46%
Приладобудівний факультет	74	330	76	4.5	103%	80	592	80	7.4	100%	100	477	100	4.8	100%	110	617	106	5.6	96%
Інженерно-хімічний факультет	48	205	48	4.3	100%	60	259	60	4.3	100%	52	466	50	9.0	96%	57	211	44	3.7	77%
<b>КПІ</b>	<b>187</b>	<b>1546</b>	<b>194</b>	<b>8.3</b>	<b>104%</b>	<b>220</b>	<b>1494</b>	<b>182</b>	<b>6.8</b>	<b>83%</b>	<b>232</b>	<b>1552</b>	<b>218</b>	<b>6.7</b>	<b>94%</b>	<b>252</b>	<b>1387</b>	<b>220</b>	<b>5.5</b>	<b>87%</b>
				<b>10.8</b>	<b>104%</b>				<b>7.3</b>	<b>82%</b>				<b>7.0</b>	<b>89%</b>				<b>5.2</b>	<b>80%</b>

Рис. 2.2. Таблиця до завдання 2.1.1

Завдання 2.1.2. Способи адресації.

Створити таблицю множення (рис. 2.3). Результати є розрахунковими. Таблицю створити не поелементним заповненням комірок, а «розтягненням» та копіюванням стовпців та рядків.

	1	2	3	4	5
1	1	2	3	4	5
2	2	4	6	8	10
3	3	6	9	12	15
4	4	8	12	16	20
5	5	10	15	20	25
6	6	12	18	24	30

Рис. 2.3. Таблиця до завдання 2.1.2

Завдання 2.1.3. Створити таблицю результатів та графік функції:

$$а) y = \begin{cases} e^x, & x < 0 \\ \cos(x), & 0 \leq x < \frac{\pi}{2} \\ x - \frac{\pi}{2}, & x \geq \frac{\pi}{2} \end{cases} \text{ для } x \text{ від } -2 \text{ до } 4 \text{ з кроком } 0.25$$

$$б) y = \begin{cases} \frac{-x}{|x|+1}, & x < 0 \\ \sin(x), & x \geq 0 \end{cases} \text{ для } x \text{ від } -1 \text{ до } 1 \text{ з кроком } 0.2$$

Перший рядок таблиці повинен містити значення аргументу, другий – функції. Файл зберегти.

#### Завдання 2.1.4. Оформлення аркушів книги.

Створити таблицю розрахунку трикутника (рис. 2.4). Створити кнопку керування видимістю рядків результатів групуванням. Звернення до полів введення даних: сторін «a», «b», кута « $\alpha$ » забезпечити не по адресу комірки, а по відповідній назві комірки «a, b, alfa». Забезпечити перевірку введення невірних даних вбудованими засобами.

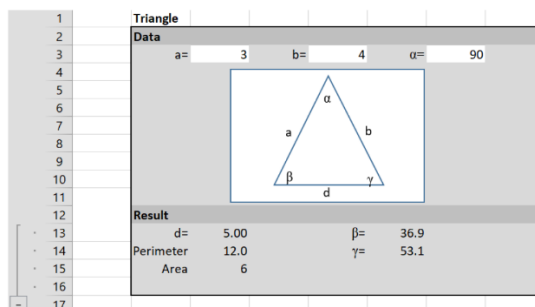


Рис. 2.4. Таблиця до завдання 2.1.4

## Практикум 2.2. Обмін даними в MS Excel

### 2.2.1. Теоретичні положення

Для розв'язання науково-технічних задач часто виникає необхідність обміну даними, зокрема імпорт даних, між *MS Excel* та іншими застосунками.

Керування імпортом даних в проводиться з групи **Data-Get External Data** стрічки. Підтримується імпорт файлів баз даних, файлів стандарту *xml*, текстових файлів.

Імпорт текстових файлів проводиться в діалоговому режимі кнопкою **Data-Get External Data-From Text** стрічки.

На першому екрані діалогу проводиться вибір типу розділювача *Delimited*, *Fixed width*, номер рядка, з якого буде почато зчитування *Start import at row*, формат кодування файлу *File origin*.

На другому екрані визначається розділювач даних *Tab*, *Semicolon*, *Comma*, *Space* та розділювач рядків *Text qualifier*.

На третьому екрані визначається формат даних для комірок таблиці *General, Text, Date*.

Дані в таблицях можна відсортувати або відфільтрувати.

Фільтрування полягає в показі на екрані тільки тих комірок, які вдовольняють якомусь критерію.

Сортування рядків чи стовпців таблиці полягає в зміні порядку їхнього взаємного розташування.

Дії сортування та фільтрація проводяться тільки з виділеним діапазоном комірок.

Керування сортуванням та фільтрацією проводиться кнопками групи **Data-Sort&Filter** стрічки

Сортування може здійснюватися в порядку зменшення або в порядку збільшення значень.

Запускається сортування кнопкою **Sort** стрічки. Кнопка активує діалогове вікно, в якому визначається стовпець *Column*, по даних якого буде проводитися сортування, ознака даних, за якою буде проводитися сортування *Sort On: Values, Cell Color, Font Color*, порядок сортування *Order*.

Кнопки **Add level, Delete Level** дозволяють додати, прибрати додаткові умови сортування.

Фільтрація запускається кнопкою **Filter** стрічки. При цьому на комірках першого рядка виділеного діапазону з'являється позначки у вигляді трикутника. Натискання позначки виводить на екран вікно, в якому можна обрати з наведеного списку значення для фільтрації, задати логічну умову (=, <, > тощо) порівняння значень комірок з якимось порогом.

Більш складні критерії фільтрації можна задати опцією *Custom Filters*, яка виводить діалогове вікно для визначення критеріїв із застосуванням логічних дій «AND» та «OR» з кількома значеннями комірок стовпця.

Треба мати на увазі, що комірки, які не виводяться на екран за рахунок фільтрації, беруть участь в розрахунках, для яких аргументи задаються діапазонами, які вміщують сховані комірки.

## 2.2.2. Завдання для самостійного виконання

Завдання 2.2.1. Робота із зовнішніми даними. Адресація сторінок.

Створити документ (рис. 2.5):

Перший аркуш «total\_list» книги містить список абітурієнтів, який завантажується з файлу «abitur.txt». Поля рядка в файлі розташовані наступним чином: номер по порядку, прізвище, пріоритет, бали, ознака вступу: 1 – поступив, 0 – не поступив, -1 – відмовився;

Другий аркуш «student\_list» книги містить список тих, хто поступив. Формується сортуванням даних зі сторінки «total\_list» з подальшим видаленням рядків тих, хто не поступив.

Третя аркуш «total» книги містить розрахунок статистики по заявах. Використовує дані зі сторінки «students».

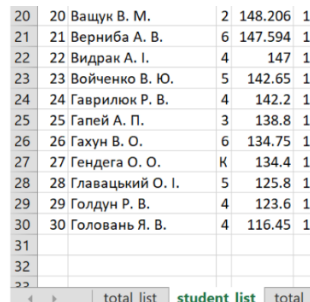
Четвертий аркуш «students» містить розрахунок статистики по абітурієнтах. Використовує дані зі сторінки «students».

Файл зберегти.



10	10 Біденко О. С.	4	142.65	0
11	11 Білоусов О. С.	К	151.4	0
12	12 Більчук Н. О.	1	155.601	1
13	13 Близнюк І. В.	К	147.594	0
14	14 Бородавко В. В.	7	134.4	0
15	15 Брень Ю. А.	6	148.95	0
16	16 Бураков М. М.	4	153.35	0
17	17 Бутенко В. Є.	6	138.8	0
18	18 Бушинський Б. А.	3	153.918	0
19	19 Ваколюк К. В.	3	174.3	0
20	20 Ващук В. М.	2	174.879	0
21	21 Верниба А. В.	6	156.4	0
22	22 Видрак А. І.	4	134.75	1
23	23 Войченко В. Ю.	5	165.3	0
24	24 Гаврилюк Р. В.	4	142.2	1

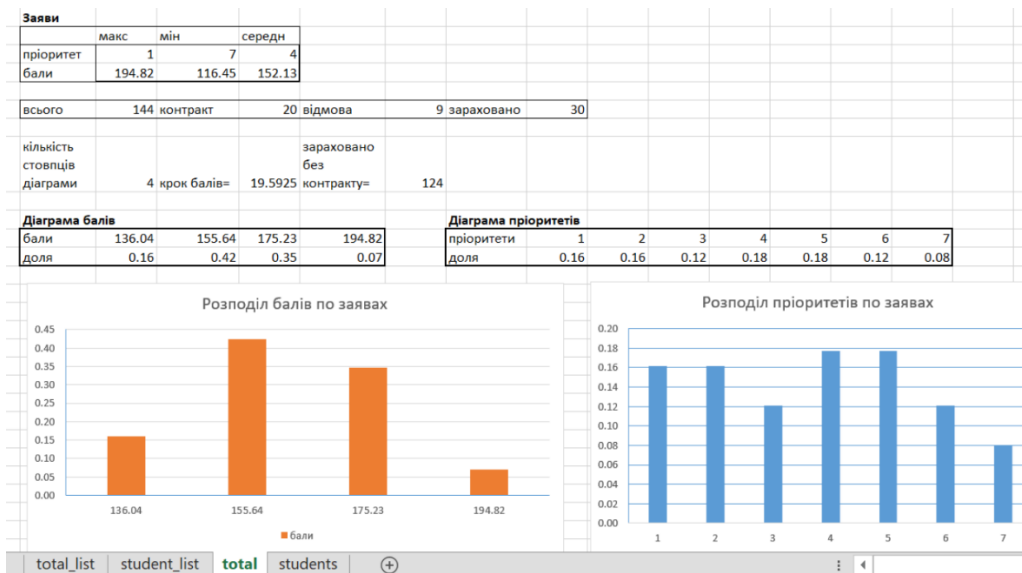
а



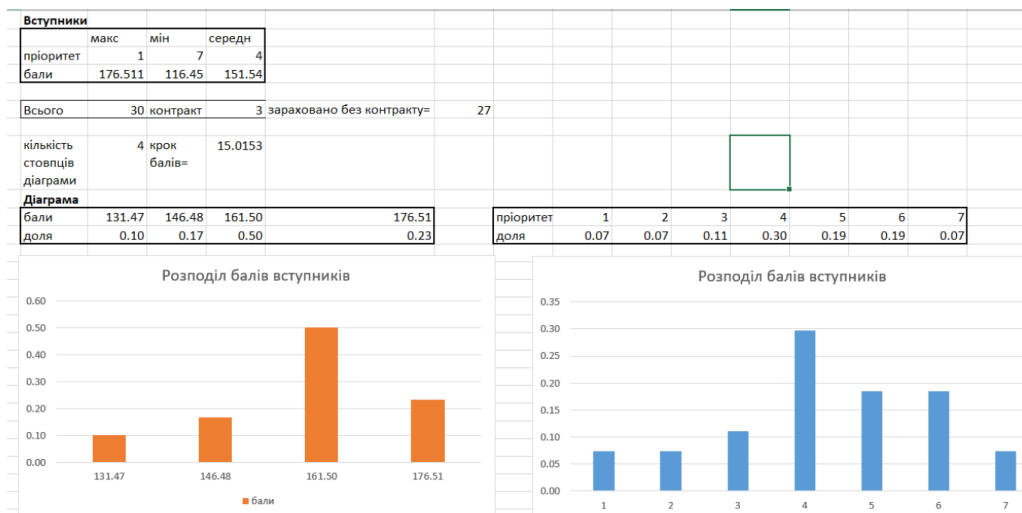
20	20 Ващук В. М.	2	148.206	1
21	21 Верниба А. В.	6	147.594	1
22	22 Видрак А. І.	4	147	1
23	23 Войченко В. Ю.	5	142.65	1
24	24 Гаврилюк Р. В.	4	142.2	1
25	25 Гапей А. П.	3	138.8	1
26	26 Гахун В. О.	6	134.75	1
27	27 Гендега О. О.	К	134.4	1
28	28 Главацкий О. І.	5	125.8	1
29	29 Голдун Р. В.	4	123.6	1
30	30 Головань Я. В.	4	116.45	1
31				
32				
33				

б





**В**



**Г**

Рис. 2.5. Книга до завдання 2.2.1: а – список абітурієнтів; б – список студентів; в – статистика по абітурієнтах; г – статистика по студентах

## Практикум 2.3. Макроси MS Excel

### 2.3.1. Теоретичні положення

#### Макроси

*Макросом* називається записана послідовність команд або дій користувача на клавіатурі. Наприклад, форматування нової таблиці,

готування таблиці до друку хованням комірок, які не потрібно друкувати, тощо. Їх можна назвати *макросами-послідовностями*.

Макроси-послідовності можуть бути записані або розроблені.

Записуються макроси-послідовності вбудованим макрорекордером, який сприймає послідовність дій користувача, переводить їх в код *VBA* та зберігає.

Розроблення макросів користувач проводить написанням програми у середовищі вбудованої мови програмування *VBA*. Програмуванням створюються макроси-функції та макроси-послідовності.

Зберігаються обидва типи макросів у вигляді програм мовою *Visual Basic for Application, VBA*.

Основне керування макросами проводиться з групи **Developer-Code** стрічки.

Для запису макросу-послідовності слід натиснути кнопку початку запису **Developer-Code- Record Macro**. В активованому вікні слід ввести ім'я макросу, місце зберігання, коментар, «гарячі» клавіши запускання макросу.

Макроси, для яких вказане місце зберігання *This Workbook*, є доступними тільки коли робоча книга, в якій вони створювалися, є відкритою в *MS Excel*. Зберігаються такі макроси в самій робочій книзі.

Макроси, для яких вказане місце зберігання *Personal Macro Workbook*, є доступними в будь-яких робочих книгах на тому комп'ютері, на якому вони створювалися. Зберігаються такі макроси в файлі *Personal.xlsb* в папці «..\AppData\Roaming\Microsoft\Excel\XLSTART\». Він завантажується автоматично під час відкриття застосунка *MS Excel*.

Для завершення процесу запису макросу слід натиснути кнопку **Developer-Code-Stop Record Macro** стрічки.

Для того, щоб починати дії макроса-послідовності з поточної комірки на момент запуску, слід перед записом макросу активувати поле **Use Relative References** стрічки.

Запуск макросу-послідовності може проводитися «гарячими» клавішами макросу або кнопкою **Macros**. Вона знаходиться в групах **Developer-Code** та **Home** стрічки. Кнопка відкриває вікно макросів, в якому можна запустити обраний з наведеного списку макрос, змінити місце зберігання макросу, видалити макрос, подивитись текст макросу.

Зручніше запускати макрос кнопками з панелі швидкого запуску, стрічки або безпосередньо на аркуші робочої книги.

Кнопка є елементом управління. Вставляється вона в аркуш полем **Developer-Controls-Insert-Form Controls-Button** стрічки.

Відразу після вставлення кнопки в аркуш на екран виводиться вікно макросів, в якому можна визначити макрос, який буде прив'язаний до кнопки. Налаштувати властивості кнопки, прив'язку макросу в тому числі, можна відповідними пунктами контекстного меню кнопки.

Прив'язка макросу до кнопки панелі швидкого доступу або стрічки проводиться у вікні налаштувань, яке відкривається полем **File-Options-Customize Ribbon/Quick Access Toolbar** стрічки або відповідними пунктами контекстного меню панелі швидкого доступу/стрічки.

Для введення кнопки слід обрати в полі **Choose command from** значення *Macros* та перетягнути потрібний пункт списку макросів, який виводиться в лівій частині вікна під полем, на праву частину вікна в визначене місце стрічки/панелі швидкого доступу.

Макроси-функції потребують використання середовища *VBA*.

Оболонка *VBA* запускається полем **Visual Basic** стрічки.

У вікні середовища відображається головне меню, дерево проєктів, вікно властивостей об'єктів, займає редактор програм (рис. 2.6).

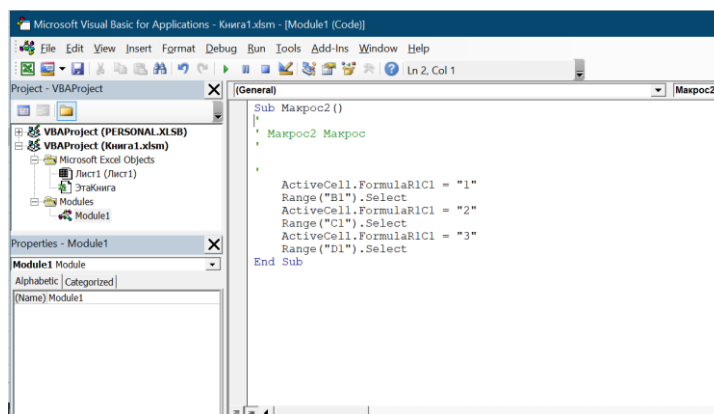


Рис. 2.6. Оболонка VBA

Компоненти програм містяться в модулях. Додавання модулів проводиться з головного меню або контекстного меню проєкту пунктом **Insert-Module**.

Введення шаблону нової підпрограми в обраний модуль можна провести пунктом **Insert-Procedure** меню в полі редактора.

В діалоговому вікні вводиться ім'я компонента, зі списку обирається тип компонента (для функції **Function**). В списку **Scope** визначається доступність компонента ( для доступних в *MS Excel* – **Public**).

В шаблон підпрограми в круглих дужках через кому вводяться імена змінних, які слугуватимуть вхідними параметрами функції. Між службовим словами *Public Function* та *End Function* функції вставляються необхідні рядки програмного коду.

За замовчанням програми зберігаються в тому файлі *MS Excel*, з якого відкривалося середовище.

Розроблений макрос-функція стає доступним разом із вбудованими функціями в менеджері функцій.

Для макросу-функції за замовчанням в менеджері функцій відсутній опис. Додати його можна у вікні макросів як **Description** через кнопку **Options**.

Перенести функцій користувача на інші комп'ютери можна як *надбудову (Add-ins) MS Excel* в файлі з розширенням *.xlam*.

Під'єднання надбудови проводиться кнопкою **File-Option-Add-ins-Manage-Excel Add-ins...** стрічки. В діалоговому вікні надбудов

слід активувати потрібну надбудову. Список надбудов за замовчанням зчитується з папки «..\Users\xx\AppData\Roaming\Microsoft\AddIns». В разі відсутності надбудови в списку слід додати файл надбудови пошуком кнопкою **Browse**.

### 2.3.2. Завдання для самостійного виконання

#### Завдання 2.3.1. Макроси

Записати в відкриту книгу макроси:

- а) починаючи з комірки В1 виводить нумерований список груп потоку
- б) прив'язати запуск макросу п.2.3.1.а до кнопки «Список», яку розташувати на аркуші;
- в) прив'язати запуск макросу п.2.3.1.а до кнопки на панелі швидкого доступу
- г) починаючи з обраної користувачем комірки виводить список груп потоку

#### Завдання 2.3.2. Макроси

Записати у відкриту книгу макрос, який в обрану комірку виводить прізвище, ім'я, по батькові шрифтом Times New Roman кеглем 14, нижче – поточну дату та час шрифтом Arial кеглем 12 червоним кольором.

#### Завдання 2.3.3. Макроси-функції

Записати у відкриту книгу функції користувача ARCCTG для розрахунку арктангенса, M\_EXP для розрахунку експоненційної функції (рис. 2.7):

$$\operatorname{arcctg}(x) = \frac{\pi}{2} - \operatorname{arctg}(x)$$
$$e^x = \sum_{n=1}^{\infty} \frac{x^n}{n!} = 1 + x + \frac{x^2}{2} + \dots + \frac{x^n}{n!} + \dots$$

```

Public Function ACTAN(x As Double) As Double
Dim Pi As Double
    Pi = 4 * ATN(1)
ACTAN = (Pi / 2 - ATN(x)) * 180 / Pi
End Function

```

```

Public Function ACTAN(x)
    Pi = 4 * ATN(1)
    ACTAN = (Pi / 2 - ATN(x)) * 180 / Pi
End Function

```

```

Public Function M_EXP(x, N)
    s = 0
    p = 1
    For i = 0 TO N
        s = s + p
        p = p * x / (i + 1)
    Next i
    M_EXP = s
End Function

```

Рис. 2.7. Програми-функції до завдання 2.3.3

### 3. СКМ MATHCAD

Програмне вікно *MathCAD* до 15-ї версії має стандартне оформлення *Windows* програм (рис. 3.1). Програмне вікно містить наступні елементи: рядок заголовка, головне меню, панелі інструментів, робочу область, рядок стану. Версії *MathCAD Prime* мають інтерфейс на базі елементів типу стрічка (*ribbon*).

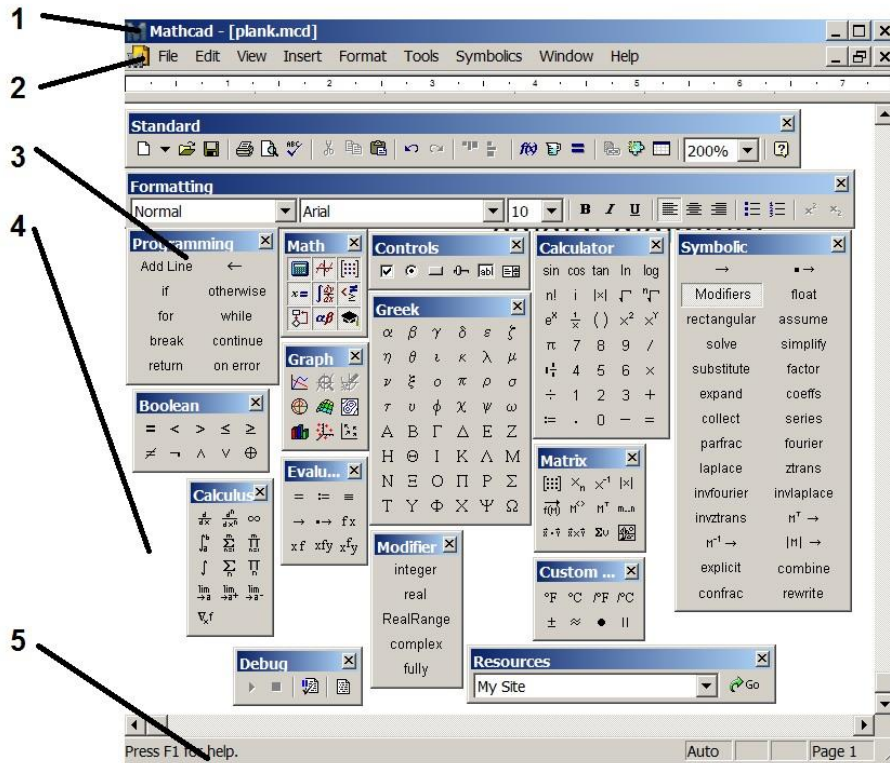


Рис. 3.1. Програмне вікно *MathCAD*: 1 – рядок заголовка; 2 – головне меню; 3 – панелі інструментів; 4 – робоча область; 5 – рядок стану

В нижній частині програмного вікна знаходиться *рядок стану*. В полях рядка відображається інформація про режими роботи пакета та документ, що редагується.

В верхній частині вікна нижче рядка заголовка знаходиться меню. Меню має наступні пункти: **File**, **Edit**, **View**, **Format**, **Tools**, **Symbolic**, **Window**, **Help**.

**Format** – керування виглядом елементів документа в робочому вікні;

**Tools** – керування режимами обчислень (**Calculate**), анімації (**Animation**), налагодження програм (**Debug**) та значеннями системних змінних точності, форматів даних, матриць;

**Symbolics** – визначення символічних операцій та виглядом результатів;

Пункт **View** меню дозволяє керувати виглядом:

**Toolbars** – панелей, які можна вивести у вікно документа або сховати;

**Ruler** – мірильної лінійки згори у вікні документа;

**Status bar** – рядка стану внизу вікна документа;

**Trace Window** – вікна трасування програм;

**Header and Footer** – колонтитулів документа;

**Regions** – виділенням кольором існуючих в документі областей;

**Annotations** - приміток у вигляді кольорових дужок біля кожної області документа;

**Refresh** – оновленням зображення документа;

**Zoom** – масштабом зображення документа.

Для виклику довідникової системи достатньо натиснути клавішу *F1*.

Якщо курсор встановлено на елемент документа, то натискання клавіші *F1* відкриває вікно довідникової системи з інформацією саме про цей елемент.

Для введення в документ *MathCAD* шаблонів математичних знаків, конструкцій програмування, графічних областей тощо, зручно використовувати *кнопкові панелі* (рис. 3.1). Керують видимістю панелей пункти меню **View – Toolbars** пакета. Кнопкову панель можна розмістити в будь-якому місці вікна *MathCAD*.

Кнопки виведення на екран більшості панелей знаходяться на панелі **Math**. Призначення кнопкових панелей пояснено в таблиці 3.1.

Панелі **Standard, Formatting, Controls, Modifier, Debug, Custom Characters** викликати можна тільки через меню **View - Toolbars**.



Таблиця 3.1. Кнопкові панелі *MathCAD*

Значок кнопки	Призначення панелі	Значок кнопки	Призначення панелі
	<i>Calculator.</i> Введення знаків арифметичних операцій, цифр		<i>Boolean.</i> Введення знаків логічних дій
	<i>Matrix.</i> Введення матриць, дії з матрицями		<i>Calculus.</i> Введення знаків алгебраїчних дій
	<i>Graph.</i> Побудова графіків		<i>Symbolic.</i> Проведення обчислень в символічному вигляді
	<i>Programming</i> Опис підпрограм-функцій		<i>Evaluation.</i> Введення знаків обчислення
	<i>Greek.</i> Введення літер грецького алфавіту		

Панелі **Standard**, **Formatting** (рис. 3.2) нагадують відповідні панелі текстового процесора *Word* та табличного процесора *Excel*.

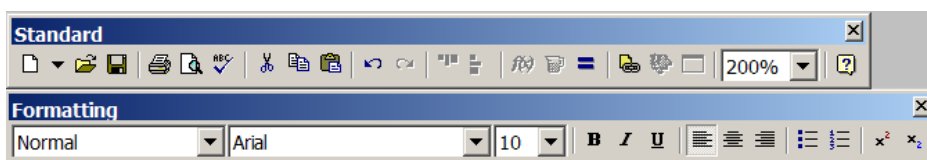


Рис. 3.2. Панелі *Standard*, *Formatting*

Панель **Controls** вміщує органи керування для створення візуального діалогу введення даних користувачем. Панель **Modifier** слугує для тонкого налаштування типів даних при символічних перетвореннях. Панель **Debug** слугує для налагодження програм. Панель **Custom Characters** містить символи, які часто використовуються: градуси Фаренгейта, Цельсія, плюс-мінус тощо.

Для виклику на екран потрібної панелі достатньо клацнути лівою кнопкою миші на значку, а потім прийомом «перетягнути-і-залишити» (*drag and drop*) розмістити панель в зручному для роботи місці вікна *MathCAD*.

Для вставки потрібної конструкції з панелі у вікно слід клацнути мишею на відповідному значку панелі.

*Робоче поле MathCAD* знаходиться під панелями інструментів. В ньому розташовуються одне чи декілька вікон документів *MathCAD*. Документ відображає математичну, текстову, графічну інформацію, яку вводить користувач та результати розрахунків системи. Інформація в документі групується в *математичних, текстових областях, областях графіків та оформлення (regions)*. *Області* в документі можуть розташовуватися в будь-якому порядку та кількості без обмежень

*Математична область* призначена для введення конструкцій пакета, необхідних для реалізації алгоритму розв'язання задачі: змінних, констант, формул, функцій, програм. Курсор в математичній області має форму червоного хреста.

*Текстова область* може містити будь які символи (кириличні, латинські) у вигляді коментарів, пояснень тощо. Робота в текстовій області здійснюється точно так, як в звичайному текстовому редакторі.

В *області графіків* виводяться двомірні та тримірні графіки.

*Область оформлення* використовується для розміщення в ній бітових графічних зображень в стандартному форматі *.jpg, .bmp*. Для оформлення документа слід вставляти зображення через *Clipboard*. В протилежному випадку навколо зображення буде зображуватися рамка з службовою інформацією. Для отримання можливості в подальшому отримати дані про зображення та редагувати його в документі його слід вставляти пунктом меню **Insert-Picture**.

Справа на робочому полі знаходиться вертикальна зелена риска. Вона розділяє документ на стовбці. Ширина стовбцю відповідає ширині аркуша. Область документа друкується по стовбцях згори вниз. По досягненню кінця інформації одного стовбцю, система переходить до друкування сусіднього, який знаходиться праворуч.

Створення документа *MathCAD* складається з введення та редагування математичних формул, графіків, визначення форматів відображення змісту документа, введення текстових описів тощо. Введення інформації в документ *MathCAD* проводиться в визначеній області.

Виділену рамкою область можна переносити в інше місце. Для цього треба клацнути мишею на області, підвести курсор до рамки. При цьому курсор перетворюється на зображення руки. Якщо до початку зсуву натиснути та тримати клавішу *Ctrl*, область буде скопійована зі збереженням її копії на первісному місці.

Аналогічним способом можна змінити розмір текстових та графічних областей. Для цього слід потягнути курсором у вигляді стрілки маркер рамки області з натиснутою лівою кнопкою миші.

В *робочому полі MathCAD* зазвичай знаходяться кілька областей. При їх зсуві чи масштабуванні можливо їхнє перекриття. Для того, щоб візуально виділити межі областей призначено пункт меню **View-Regions**. Всі проміжки між областями зафарбовуються в сірий колір так, що світлі області стають виділеними на темному фоні.

Пункт меню **Format-Separate Regions** автоматично зсуває області таким чином, що вони розташовуються без перекриття.

Для горизонтального чи вертикального вирівнювання областей призначені пункт меню **Format-Align Regions**. Декілька областей групуються щигликом миші з утриманням клавіші *Shift* або рамкою виділення мишею. Далі, в залежності від вибору: **Across**, **Down**, - вони вирівнюються відповідно по горизонталі чи вертикалі.

Властивості областей можна редагувати за допомогою контекстного меню. Контекстне меню викликається клацанням правої кнопки миші на області. Зміст меню визначається типом області.

При запуску системи за замовчанням встановлюється математична область введення.

Для обмеження доступу до інформації області можуть бути закриті паролем (*lock*) або сховані (*collapse*), тобто прибрані з екрану.

Керування (*lock*, *collapse*) областями можливо, коли вони поєднані в *area*. *Area* займає весь рядок документа та вміщує в себе всі області, які знаходяться всередині неї. Вводиться в документ пунктом меню **Insert-Area**.

Для прибирання областей з екрану можна застосувати три способи.

Спосіб 1. Слід два рази клацнути на верхньому маркері *Area*. Повторне клацання виводить зміст *Area* на екран.

Спосіб 2. Пунктами контекстного меню **Area-Collapse/Expand**.

Спосіб 3. Пунктами меню **Format-Collapse/Expand**.

Для блокування доступу до *Area* можна застосувати пункти **Lock**, **Unlock** контекстного меню верхнього маркера *Area* або пункти меню **Format-Lock/Unlock**.

Введення команд може бути виконано декількома способами:

- шляхом натискання комбінацій «гарячих» клавіш;
- вибором пункту меню або кнопки однієї з панелей;
- написанням команди безпосередньо в тексті документа.

### **Робота з текстовою областю документа**

*Текстова область* може бути створена трьома способами. В усіх способах спочатку треба встановити курсор в потрібне місце документа.

Перший спосіб є найпростішим: треба просто почати вводити текст. Коли після першого слова буде введено пробіл, математична область перетвориться в текстову. Текст буде взято в прямокутну рамку, яка відображає текстову область.

Другий спосіб полягає в тому, що введення треба почати з символу лапок ". З'явиться прямокутна рамка текстової області.

Третій спосіб – застосувати пункт меню **Insert-Text Region**.

Текстова область розширюється при введенні тексту. Щоб завершити введення тексту, необхідно клацнути мишею поза текстовою областю. Для редагування текстової області необхідно клацнути на ній мишею. Ознакою режиму редагування є курсор у вигляді червоної вертикальної лінії.

Для текстової області можна задавати тип шрифту, його розмір, стиль, колір, параметри абзацу, колір фону. Визначення можна проводити за допомогою головного меню **Format-Text**, панелі **Formatting**, контекстного меню.

Корисною можливістю, яка забезпечує наочність документа є можливість вставляння математичної області безпосередньо всередину текстової області. Для цього слід поставити курсор редагування текстової області в потрібне місце та використати пункт **Insert Math Region** контекстного меню текстової області.

## Збереження документа MathCAD

Для збереження документа *MathCAD* в файлі необхідно застосувати пункт меню **File-Save as**.

Документи мають власний формат. Пакети старших версій можуть читати документи, які розроблені пакетами молодших версій. До версії 12 документи мали розширення *.mcd*. За впливом компанії *Microsoft* далі документи було переведено у стандарт *xml* та вони отримали розширення *.xmcd*.

Зберегти документ (рис. 3.3) можна не тільки в форматі *MathCAD* з розширеннями *.xmcd*, *.mcd*, а також в форматі *web .htm* та форматі *RTF .rtf*.

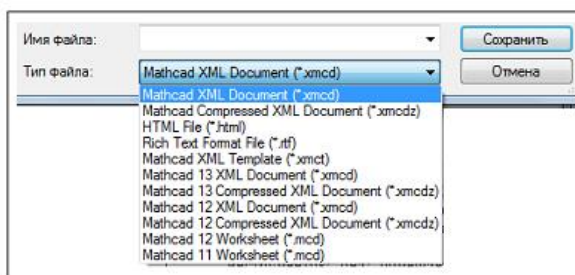


Рис. 3.3. Список форматів документа *MathCAD*

## Практикум 3.1. Загальні обчислення

### 3.1.1. Теоретичні положення

#### Організація обчислень

В системі передбачено засоби для роботи з наступними типами даних:

- цілі ( 12, -24, 0 тощо);
- дійсні. Можуть записуватися в формі фіксованої крапки (3.265) та в експоненційній формі ( $a \cdot 10^{-p}$ , де  $a$  – ціле або дійсне число з фіксованою точкою – мантиса,  $p$  – десятковий порядок);

- комплексні ( $a + bi$ , де  $a$  – дійсна частина,  $b$  – уявна,  $i/j$  – уявна одиниця). Літера уявної одиниці визначається пунктом **Format-Result-Display options-Imaginary Unit** меню (рис. 3.4). Між уявною частиною та уявною одиницею знак множення НЕ СТАВИТЬСЯ;
- рядки – будь яка послідовність символів, що забрана в лапки («Це рядок»);
- одиниці вимірів фізичних величин.

Особливим типом даних є **NaN** (*NotANumber*). Він призначений для ідентифікації елементів масивів, в які не занесені жодні дані. Наприклад, при зчитуванні з зовнішнього файлу матриці, пропущеним елементам буде надано тип **NaN**. При побудові графіків елементи з типом **NaN** ігноруються.

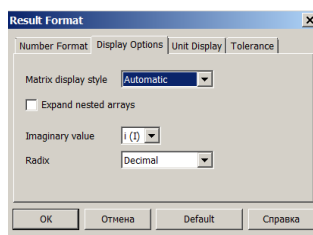


Рис. 3.4. Визначення літери уявної одиниці

Дані використовуються у вигляді змінних: *простих змінних, масивів у вигляді векторів та матриць, ранжованих змінних*. Змінним може бути надана розмірність для розв'язання фізичних задач.

За зоною дії в документі змінні діляться на *локальні та глобальні*. *Локальне* визначення діє від місця визначення вниз до кінця документа, *глобальне* визначення діє на весь документ, незалежно від місця визначення.

Кожна змінна повинна мати унікальне ім'я. Ім'я змінних складається з латинських чи грецьких літер та цифр. Використання в імені кирилиці чи пробілів не дозволяється.

Для керування математичними діями в пакеті призначені кілька *системних змінних*. Визначення системних змінних можна проводити безпосередньо в робочому полі або у вікні меню **Tools-Worksheet Options-Built-in-Variables** (рис. 3.5).

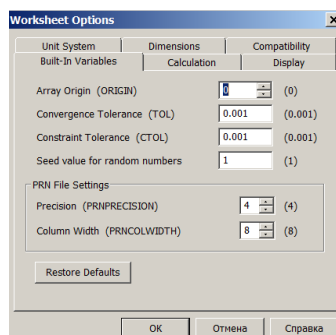


Рис. 3.5. Вікно системних змінних

**ORIGIN** – визначає початковий індекс масивів. Значення за замовчанням – 0.

**TOL** – визначає точність чисельного обчислення рівнянь, інтегралів, похідних. Значення за замовчанням 0.001.

**CTOL** – визначає точність чисельного обчислення систем рівнянь блоків **Given**. Значення за замовчанням 0.001.

**PRNPRECISION**, **PRNCOLWIDTH** визначають формат запису даних в текстові файли ASCII функціями **WRITEPRN**, **APPENDPRN**. Значення за замовчанням 4, 8, відповідно.

Примітка. При повторному застосуванні імені, яке вже використовується системою, ім'я підкреслюється зеленою хвилястою лінією. Така ситуація можлива, коли користувач вводить змінну, яка в системі визначена, як фізична константа тощо. Наприклад,  $c$  – швидкість світла  $2.998 \cdot 10^8 \text{ м} \cdot \text{с}^{-1}$ .

Грецькі літери вводяться з панелі *Greece* або комбінацією «латинська літера + *Ctrl* +G (табл. 3.2).

Таблиця 3.2. Клавіатурна відповідність літер грецького алфавіту

Буква	$\alpha$	$\beta$	$\delta$	$\varepsilon$	$\varphi$	$\Gamma$	$\Phi$	$\lambda$	$\eta$	$\Omega$	$\pi$	$\theta$	$\rho$	$\sigma$	$\tau$
Клавіша	a	B	d	e	F	G	H	l	n	O	p	q	r	s	t

В пакет вбудовано кілька констант (табл. 3.3):

Таблиця 3.3. Вбудовані константи

Назва	Знак	Введення	Значення
Основа натурального логарифму	e	e	2.718
Нескінченність	$\infty$	Ctrl+Shift+z	$1 \cdot 10^{304}$
Квадратура кола	$\pi$	Ctrl_Shift+p	3.141592
Відсоток	%	%	0.01

Для візуального ефекту ім'я може мати підрядковий індекс (підрядковий індекс НЕ Є ЧИСЛОМ ІНДЕКСУ МАСИВУ). Підрядковий індекс вводиться за допомогою клавіші «.» (точка).

Наприклад, послідовність «C+.i» дає змінну C<sub>i</sub>, де «i» не є індексом, а є часткою імені.

Великі та малі літери в іменах визначають різні імена.

Дані цілого типу можуть оброблятися в десятковій, восьмеричній, двійковій та шістнадцятиричній системах числення. За замовчанням використовується десяткова система. Для явного визначення системи числення слід після чисельного значення вказати літеру «o» для восьмеричної системи, «h» – для шістнадцятиричної, «b» – двійкової. Визначення системи числення результату проводиться пунктом **Format-Result- Display options – Radix** меню.

Для визначення змінної застосовується знак прирівнювання. В системі розрізняють *локальне та глобальне* прирівнювання. *Локальне* «:=» прирівнювання розповсюджується вправо-вниз від місця визначення. *Глобальне* прирівнювання «≡» діє на увесь документ незалежно від місця визначення. Глобальне прирівнювання доцільно застосовувати для констант, які передбачається використовувати у всьому документі.

Для введення знаку локального прирівнювання слід використати панель *Evaluate* або ОДНУ клавішу двокрапки «:». Наприклад,

**Ім'я змінної := Вираз**

Кількість прирівнювань однієї змінної в документі не обмежується.



Для введення знаку глобального порівняння слід використати панель **Evaluate** або клавішу «~». Наприклад,

**Ім'я змінної**  $\equiv$  **Вираз**


Для виведення значень змінних та обрахунків призначено знак рівняння «=». Для введення знаку рівняння слід використати панель **Evaluate** або клавішу «=».

Знаки рівняння та порівняння можна використовувати ланцюгом. Наприклад, запис  $c := \sqrt{5} =$  присвоює змінній  $c$  значення обрахованого виразу та виводить на екран значення 2.236

### Введення математичних виразів

Обчислення проводяться в *математичній області*. *Математична область* встановлюється за замовчанням. Ознакою *математичної області* є курсор у вигляді червоного хреста. З цього місця можна починати введення даних. З початком введення курсор набуває вигляд блакитного кута.

Кут може мати праву та ліву орієнтацію. Орієнтація показує напрям введення виразу. Зміна орієнтації проводиться натисканням клавіші *Insert*, зсув маркера по виразу – клавішами стрілок «→», «←», зміна горизонтального розміру маркера – клавішею *Space*.

Наприклад, якщо кут правий, то наступний символ «+» вводиться праворуч: .

Якщо кут лівий, то символ вводиться зліва: .

Фрагмент, який горизонтально охоплює кут є операндом для операції чи функції. При введенні арифметичного знаку, дія якого відноситься до фрагменту математичного виразу, необхідно змінити розмір кута так, щоб він охоплював фрагмент.

Знаки арифметичних операцій вводяться з клавіатури або кнопками панелі **Calculator**. Деякі математичні операції наведено в таблиці 3.4.

Таблиця 3.4. Введення основних арифметичних операцій з клавіатури

Дія	Зображення	Клавіші	Дія	Зображення	Клавіші
-----	------------	---------	-----	------------	---------

додавання	2+3	+	модуль	3	
віднімання	2-3	-	корінь	$\sqrt{3}$	\
ділення	$\frac{2}{3}$	/	факторіал	2!	!
множення	2·3	*	n-й корінь	$n\sqrt{3}$	Ctrl+\
ступінь	X2	^	комплексне пов'язання	$\overline{2+3j}$	Shift +“
додавання в два рядки	$\begin{matrix} 2 \dots \\ +3 \end{matrix}$	Ctrl+Enter	ділення в один рядок	2I3	Ctrl+/ Ctrl+

Всі результати та проміжні дані зберігаються в пам'яті системи з максимальною точністю 15 знаків мантиси. Вигляд результату може визначати користувач в діалоговому вікні (рис. 3.8). Виклик вікна проводиться пунктом **Format-Result-Number Format** меню або подвійним щигликом безпосередньо на результаті.

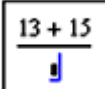
Наприклад, обрахування значення виразу  $\frac{13+15}{3 \cdot (27-8)} \cdot \frac{56-34}{19} = 0.569$ .

може складатися з наступних дій:

1) Набрати чисельник першого першого множника та клавішею *Space*

охопити його: ;

2) до виділеного виразу застосувати арифметичну операцію ділення :



3) набрати знаменник та охопити весь дріб правим кутом, щоб справа

ввести знак множення: ;

4) ввести знак множення:  $\frac{13 + 15}{3 \cdot (27 - 8)} \cdot$

5) аналогічно пунктам 1-3 провести набір другого множника та охопити весь вираз правим кутом:

$$\frac{13 + 15}{3 \cdot (27 - 8)} \cdot \frac{56 - 34}{19}$$

6) ввести оператор виведення =:  $\frac{13 + 15}{3 \cdot (27 - 8)} \cdot \frac{56 - 34}{19} = 0.569$

### Формати виведення результатів

Результати можуть бути виведені в наступних форматах *Decimal*, *Scientific*, *Engineering*, *General*, *Fraction*.

*Decimal* – формат дійсних чисел з фіксованою крапкою. Кількість цифр після коми задається в полі **Number of decimal places**. Поле **Show trailing zeros** визначає, чи додавати в кінці числа нулі до потрібної кількості цифр.

*Scientific* – експоненційний формат. Поле **Show exponents as E+000** визначає як виводити основу: у вигляді 10 або E (1.22· 10<sup>2</sup> чи 1.22E+2).

*Engineering* – експоненційний формат. Відмінність від формату *Scientific* полягає в тому, що десятковий порядок кратний 3.

*General* – універсальний формат. Якщо кількість цифр цілої частини числа не перебільшує значення з поля **Exponential threshold**, то виводиться число з фіксованою крапкою. В протилежному випадку – в експоненційній формі.

*Fraction* – формат у вигляді дроби. Поле **Use mixed numbers** дозволяє виділити цілу частину дроби.

Наприклад, формат *Fraction* для виразу  $\frac{8}{6}$  без поля **Use mixed numbers** дає результат  $\frac{4}{3}$ , з полем –  $1\frac{1}{3}$ .

Визначення формату проводиться обраним пунктом діалогового вікна (рис. 3.8).

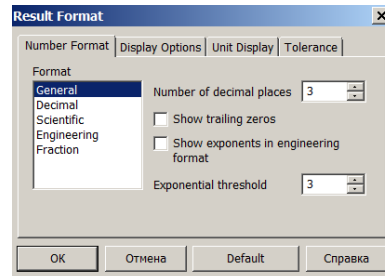


Рис. 3.8. Вікно визначення форматів результатів обчислень

В системі передбачено два режими обчислень:

- *автоматичний режим*, коли всі обчислення виконуються автоматично під час введення формул;
- *ручний режим*, коли старт обчислень окремого виразу чи всього документа задається користувачем.

Коли вираз знаходиться в процесі розрахунку чисельним або символічним процесором, він виділяється рамкою зеленого кольору, а будь-які дії користувача щодо подальшого редагування документа блокуються.

Автоматичні обчислення спрощують роботу з документом, оскільки результати розрахунків з'являються в реальному часі, і користувач має можливість аналізувати їх відразу. З іншого боку, якщо обчислення складні, вони можуть забирати багато часу. Зміна якого-небудь виразу на початку великого документа, який впливає на наступні обчислення, призводить до перерахунку всіх залежностей заново. У таких випадках часто зручніше редагувати текст в ручному режимі, а обчислення включати в міру необхідності.

*Автоматичний режим* включається за замовчанням при створенні нового документа.

В *ручному режимі* користувач повинен запускати обчислення самостійно. При редагуванні тексту в ручному режимі не виконуються ані

обчислення виразів, ані побудова графіків, відповідні місця в виразах відзначаються маркерами (рис. 3.9).

Для того щоб обчислити увесь документ слід обрати пункт меню **Tools- Calculate- Calculate Worksheet** або клавіші *Ctrl+F9*.

Для обчислення всіх формул у видимій на екрані частині документа слід обрати пункт меню **Tools- Calculate- Calculate Now** або клавішу *F9*.

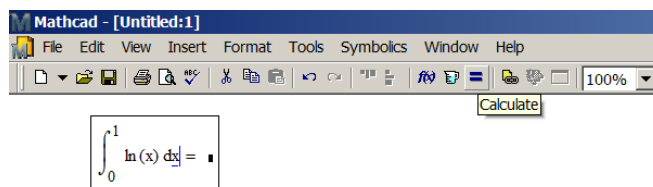


Рис. 3.9. Вигляд документа в ручному режимі обчислень

Найпростіше запустити розрахунки можна клацанням на кнопці **Calculate** із зображенням знака рівності на панелі **Standard** (рис. 3.9).

Щоб перервати тривалий процес обчислень треба натиснути клавішу *Esc*. У цьому разі вирази, які *Mathcad* не встиг вирахувати, будуть позначені в документи червоним кольором.

Перерване обчислення поновлюються натисканням клавіші *F9*.

### Оптимізація обчислень

Особливістю *Mathcad* є прискорення чисельних обчислень за рахунок застосування елементів символічної математики. Безпосередньо перед чисельним розрахунком *Mathcad* автоматично намагається спростити вираз, використовуючи символічний процесор. Це називається *оптимізацією*. Режим оптимізації вмикається для цілого документа або для окремих формул.

Щоб увімкнути або вимкнути режим оптимізації всіх виразів в документі слід застосувати пункт меню **Tools-Optimize**.

Щоб змінити режим оптимізації для окремих виразів слід виділити цю формулу лініями введення і вибрати пункт **Tools-Optimize-Equation** у меню або обрати такий пункт в контекстному меню виразу.

Ознакою того, що вираз знаходиться в режимі оптимізації є блакитний квадрат справа від виразу.


## Відключення обчислення окремих формул

*Mathcad* дозволяє відключити обчислення будь-якої формули. При цьому вона не буде впливати на подальші обчислення. Щоб не обчислювати окрему формулу в документі треба в контекстному меню виразу обрати пункт **Disable Evaluations**.

Еквівалентний спосіб відключення обчислення окремої формули полягає у виклику діалогового вікна **Properties** через однойменний пункт контекстного меню. У діалозі **Properties** слід перейти на сторінку **Calculation** і обрати **Disable Evaluations**. Про виключення формули з обчислень можна судити за наявністю чорного маркера відразу за цією формулою.

## Вбудовані функції

Для обчислення значень найбільш поширених математичних функцій в *MathCAD* застосовуються так звані *вбудовані функції* (наприклад, синус, корінь квадратний тощо). Для звернення до функції необхідно ввести її ім'я або умовний символ та аргумент.

Функцію можна ввести з клавіатури, пунктом меню **Insert-Function**. Найбільш уживані функції розташовані на панелі **Calculate**. Повний список функцій і їх синтаксис можна отримати у вікні **Insert Function** (рис. 3.10), клацнувши на кнопці  панелі інструментів **Standart**.

Весь набір вбудованих функцій у вікні **Insert Function** розбитий на групи відповідно до функціонального значення:

*Bessel* — спеціальні звичайні та модифіковані функції Беселя;

*Complex Numbers* — функції роботи з комплексними числами;

*Curve Fitting and Smoothing* — функції апроксимації кривих;

*Differential Equation Solving* — функції розв'язання диференціальних рівнянь;

*Expression Type* — функції для визначення типів даних;  
*File Access* — функції для роботи з файлами;  
*Finance* — фінансові функції;  
*Fourier Transform* — функції перетворення Фур'є;  
*Graphing* — функції для роботи з графічними об'єктами;  
*Hyperbolic* — гіперболічні функції;  
*Image Processing* — функції для обробки зображень;  
*Interpolation and Prediction* — функції інтерполяції та екстраполяції;  
*Log and Exponential* — логарифмічні функції;  
*Lookup* — функції для пошуку даних;  
*Number Theory/Combinatorics* — функції теорії чисел;  
*Piecewise Continuous* — кусково-неперервні функції;  
*Probability Density, Probability Distribution, Random Numbers* — функції теорії імовірності;  
*Signal Processing* – функції для обробки сигналів;  
*Solving* – функції розв'язання рівнянь;  
*Sorting* – функції сортування даних;  
*Statistics* – статистичні функції;  
*String* – функції для роботи з текстом;  
*Trigonometric* – тригонометричні функції;  
*Truncation and Round Off* – функції округлення;  
*User Defined* – функції користувача;  
*Vector and Matrix* – функції для роботи з матрицями.

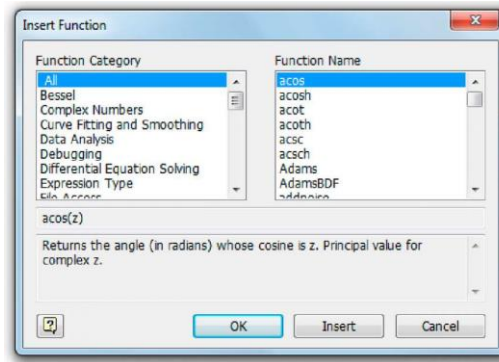


Рис. 3.10. Вікно вбудованих функцій

У вікні необхідно вибрати відповідну категорію **Function Category**. При цьому у вікні **Function Name** будуть представлені функції *MathCAD*, що відносяться до даної категорії. Якщо необхідна функція в обраній категорії не буде знайдена, можна переглянути категорію *All*, що містить всі вбудовані функції. Короткий опис виділеної функції показано в нижній частині вікна. Повний опис можна отримати з довідки, яка викликається за посиланням «?» в лівому нижньому кутку вікна.

Далі наведені найбільш уживані вбудовані математичні функції *MathCAD*:

**sin(z)**, **cos(z)**, **tan(z)**, **cot(z)**, **sec(z)**, **csc(z)** – тригонометричні функції: синус, косинус, тангенс, котангенс, секанс, косеканс.

Для зворотних тригонометричних функцій перед іменем додається літера «а». Наприклад, **asin(z)** – арксинус.

Для гіперболічних функцій в кінці імені додається літера «h». Для зворотних гіперболічних попереду додається «а», в кінці – «h». Наприклад, **sinh(z)** – гіперболічний синус, **asech(z)** – гіперболічний арксеканс.

**atan2(x, y)** –  $\text{atan2}(x,y) = \text{atan}(y/x)$

**sinc(z)** – функція  $\text{sinc}(z) = \sin(z)/z$

**exp(z)** – експоненційна функція ( $e^z$ )

**ln(z)** – логарифм натуральний z

**log(z, b)** – логарифм від z по основі b. Якщо другий аргумент «b» не визначено, основою вважається число 10.



**DMS (x)** – значення кута x в радіанах.

Аргументи тригонометричних функцій визначаються в радіанній мірі. Для позначення даних в градусах можна додати після виразу службове слово **deg**.

### Функції роботи з комплексними числами

Більшість вбудованих функцій можуть мати в якості аргументів комплексні числа, результатами обчислень таких функцій також є комплексні числа.

Для аналізу комплексних значень в системі передбачені функції обробки комплексних чисел.

Для визначення комплексного числа достатньо знати два параметри. Це можуть бути дійсна C і уявна B частини комплексного числа або його модуль A і фаза  $\varphi$ . Ці параметри взаємопов'язані.

$$z = C + jB = A \cdot e^{j\varphi} \quad A = \sqrt{C^2 + B^2} \quad \varphi = \arctg\left(\frac{B}{C}\right)$$

Функції розрахунку дійсної та уявної частин комплексного числа  $z$  в *Mathcad* мають загальноприйняті математичні назви: **Re (z)** для розрахунку дійсної частини та **Im (z)** – для уявної. В термінах *Mathcad* фаза комплексного числа називається його аргументом. Визначити аргумент можна за допомогою функції **arg (z)**.

Функція **Csgn (z)** повертає знак дійсної частини комплексного числа  $z$ . Якщо дійсна частина числа дорівнює нулю, функція повертає знак уявної частини.


Примітка. Часто вбудовані функції *Mathcad* мають деякі особливості застосування (наприклад, обмежена область визначення, обробка комплексних аргументів тощо), незнання яких може привести до помилок. Це зауваження, відноситься не тільки до *Mathcad*, а й до більшості аналогічних математичних пакетів. Якщо є хоча б найменший сумнів в адекватності обраної функції для розв'язуваної задачі, то доцільно витратити додатковий час на ознайомлення з повною довідковою інформацією по функції.


## Розширені арифметичні оператори

*MathCAD* має ряд операторів, що дозволяють спростити запис сум, добутків, похідних, визначених інтегралів, меж. Такі оператори часто називають *розширеними арифметичними операторами* або *операторами математичного аналізу*.

Ці оператори (значок оператора) зручно вводити в документ *MathCAD* використовуючи панель **Calculus**.

При введенні оператора зображується шаблон для подальшого заповнення числовими або символічними значеннями.

*Оператор обчислення суми.* Для введення оператора в документ необхідно клацнути на кнопці  та заповнити поля шаблону (рис. 3.11а).

*Оператор обчислення доданку.* Для введення оператора в документ слід клацнути на кнопці  та заповнити поля шаблону (рис. 3.11б).

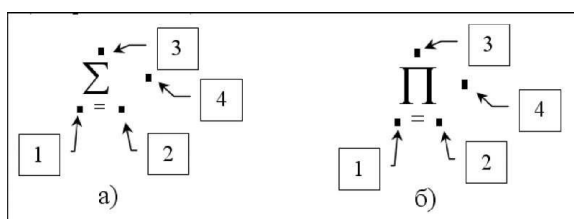


Рис. 3.11. Шаблиони операторів сум та доданків: а – шаблон сум; б – шаблон доданку [1]

1 – ім'я параметра сумування (множення); 2 – нижня межа; 3 – верхня межа; 4 – вираз, який залежить від параметра сумування (множення)

Розширені арифметичні оператори мають «гарячі» клавіші (рис. 3.12).

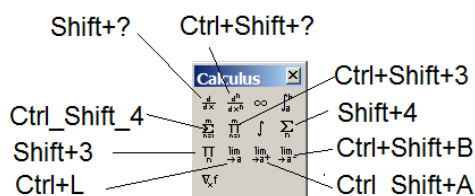


Рис. 3.12. Клавіші розширених операторів [1]

## Шаблиони виразів та функцій

Натисканням кнопки панелі **Evaluate, Calculate** на екран виводиться шаблон виразу з позначеннями відповідної дії та маркерами (рис. 3.14). В поля маркерів необхідно вставити чисельні значення, імена змінних або інші математичні символи. Це дозволяє використовувати для запису виразів окрім звичного клавіатурного порядку – шаблонний [8].

Наприклад, клавіатурний порядок введення виразу

$$\left(\frac{2}{3+4}\right)^5 = 1.904 \times 10^{-3}$$

може мати наступний вигляд:

2→/→3→+→4→ **Space** → **Space** → **Shift+6**→5→=

Такого ж результату можна отримати із застосування шаблонів.

Наприклад, застосування панелей для введення виразу

$$\frac{\ln\left(\frac{x+y}{5+y}\right) + y^2}{\sqrt[3]{x+y}} \cdot e^{-x} \quad \text{дає наступний шаблон} \quad \frac{\ln\left(\frac{\square + \square}{\square + \square}\right) + \square}{\sqrt[\square]{\square + \square}} \cdot e^{-\square}.$$

структура заповнюється за 10 кроків [8]: визначаються 7 змінних та 3 константи.

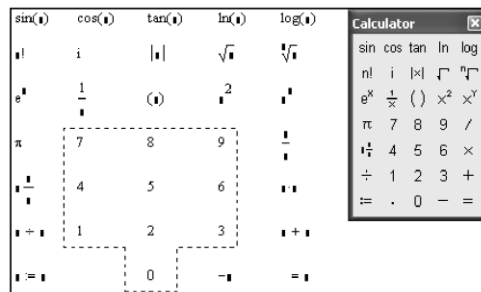


Рис. 3.14. Шаблиони з панелі **Calculator** [8]

Шаблиони дозволяють зробити дуже універсальним виклик функцій. Шаблиони функцій зосереджено на панелі **Evaluate** (рис. 3.15). Панель містить чотири шаблиони виклику функцій, відмінних від традиційного:

префіксний, постфіксний для функцій одного аргументу, інфіксний та «дерево» – для функцій двох аргументів

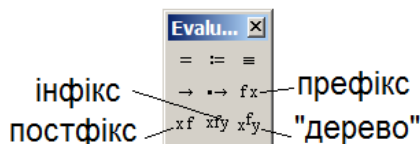


Рис. 3.15. Шаплони виклику функцій

Наприклад, записати вираз  $\arctg\left(\frac{3}{4}\right)$  можна з використанням вбудованої функції **atan2** можна

традиційним клавіатурним способом **atan2 (3, 4) ;**

префіксним способом: **fx** → ■ ■ → **atan2 0.75;**

постфіксним способом: **xf** → ■ ■ → **0.75 atan2;**

змішаним способом: **x^fy** → ■ ■ ■ → **4 atan2 3;**

у вигляді «дерева»: **x^fy** → ■ ■ ■ →  $\begin{matrix} & \text{atan2} & \\ \swarrow & & \searrow \\ 4 & & 3 \end{matrix}$ .

Не можна стверджувати, що якийсь спосіб має переваги. Обидва способи є тільки інструментами. В кожному конкретному випадку користувач має можливість використати переваги одного чи іншого, комбінуючи клавіатурне та шаблонне введення.

### Функції користувача

У *MathCAD* користувач може на додачу до вбудованих функцій створювати *функції користувача* (локальні функції). На відміну від простої змінної, значення такої функції залежить від значень аргументів, а на відміну від вбудованої функції, ця функція визначається самим користувачем. *Функції користувача* спрощують структуру документа при необхідності багаторазових розрахунків одного виразу з різними вихідними значеннями, дозволяють модифікувати вбудовані функції в разі їхньої невідповідності вимогам користувача, створювати нові функції, які адаптовано до конкретної науково-технічної тематики.

Перед тим як використовувати функцію користувача, потрібно її задати, тобто виконати *опис* [9]. *Опис функції* користувача розміщується в робочому документі перед викликом функції і включає в себе ім'я функції, список формальних параметрів і тіло функції.

Кожна функція користувача в *MathCAD* повинна мати оригінальне ім'я, по якому здійснюється звернення до цієї функції. Через це ж ім'я (і тільки через це ім'я) «повертається» в робочий документ результат виконання функції користувача.

Після імені функції розміщується список *формальних* параметрів, взятий у круглі дужки. Через формальні параметри «всередину» функції «передаються» дані необхідні для виконання обчислень всередині програми. Як формальні параметри можуть використовуватися імена простих змінних, масивів і функцій. формальні параметри відокремлюються один від одного комами.

Вираз тіла функції може вміщувати тільки один рядок.

*Визначення функції* користувача має вигляд

**<ім'я функції> (<список формальних параметрів>) : = <вираз, залежний від формальних параметрів>**

Як формальні параметри використовуються тільки імена (а не більш складні вирази) і ці параметри показують як значення функції залежить від аргументів, тобто ці параметри повинні бути присутніми і в правій частині опису функції користувача. Не має значення чи були раніше визначені або використані в робочому документі імена формальних параметрів. Імена формальних параметрів автономні в тому сенсі, що вони не співвідносяться до імен, які використовуються в документі.

Для виконання під функції користувача необхідно звернутися до імені функції з вказівкою списку *фактичних* параметрів.

*Фактичні* параметри вказують, при яких конкретних значеннях здійснюються обчислення в тілі програми. фактичні параметри відокремлюються один від одного комами. Імена фактичних параметрів при виклику функції можуть не збігатися з іменами її формальних параметрів.

Між фактичними і формальними параметрами має бути відповідність по кількості, порядку проходження і типу. Остання відповідність означає, що [9] :

- якщо формальним параметром є проста змінна, то в якості фактичного може використовуватися константа, змінна, арифметичний вираз;
- якщо формальним параметром є вектор чи матриця, то фактичним параметром повинен бути вектор або матриця;
- якщо формальним параметром є ім'я вбудованої функції чи іншої програми, то і фактичним параметром повинний бути той самий об'єкт.

Звернення до функції має відбуватися після опису функції.

Для обчислення значення функції користувача достатньо записати ім'я функції і список фактичних параметрів, взятий у круглі дужки. Фактичні вказують, при яких конкретних чисельних значеннях аргументів буде обчислюватися функція. В якості фактичних параметрів може виступати константа, число, змінна, арифметичний вираз. Очевидно, що кількість, типи і порядок запису фактичних і формальних параметрів повинен бути однаковий.

Найбільш близька за змістом до функції користувача пакета *MathCad* є функція *DEFFN* мови *BASIC*.

Наприклад, визначення функції користувача *dist (x, y)*, яка обчислює відстань між точкою з координатами (x, y) і початком координат може мати наступний вигляд:

$$dist(x, y) := \sqrt{x^2 + y^2}$$

Можливі варіанти використання функції *dist (x, y)* описано нижче:

$$x1 := 2.1 \quad y1 := -0.1$$

$$z := \frac{dist(x1, y1)}{2} \quad z = 1.051 \quad dist(-0.2, 8.1) = 8.102$$

Примітка. Функції користувача можуть мати ТІЛЬКИ один рядок тіла функції.

### **Розрахунки в діапазоні значень**

Для побудови графіків, розрахунків функцій в діапазоні значень (табуляція функцій) тощо. необхідно проводити обчислення одного виразу не один, а багато разів при різних значеннях аргументів.

Для таких випадків у пакеті передбачені засоби для розрахунків в діапазоні значень. Цей вид розрахунків реалізований за допомогою *ранжованих* змінних. В деяких джерелах вони називаються *дискретними змінними* [10, 11].

*Дискретна/ранжована змінна* – змінна, яка приймає не одне, а декілька значень. Значення ранжованої змінної змінюються за законом арифметичної прогресії. Априорі відомо початкове значення змінної, крок зміни, кінцеве значення, а значить – кількість повторень циклу.


Аналогом ранжованої змінної є операція зумовленого циклу **for** в мовах програмування високого рівня.

Відміною ранжованої змінної від циклу програмування є те, що цикл виконується по черзі для одного значення змінної циклу над рядками, а ранжований цикл виконується по черзі над рядками для всіх значень змінної.

Ранжована змінна задається в наступний спосіб:

**<ім'я змінної>:=початок, друге значення .. останнє значення**

Друге значення дорівнює сумі першого значення та кроку зміни змінної. Якщо крок змінної дорівнює одиниці, то задавати його не обов'язково.

Для введення двох крапок між другим та останнім значеннями застосовується символ «:» (двокрапка) або кнопка  панелі *Matrix*.

Наприклад, ранжована змінна  $x = 0.1, 0.3 .. 1.5$  змінюється від 0.1 до 1.5 з кроком 0.2.

Для виведення всіх значень дискретної змінної слід набрати її ім'я та поставити після нього знак прирівнювання:  $x =$

Примітка. Вивести одне значення ранжованої змінної неможливо. Присвоїти значення ранжованої змінної іншій змінній НЕМОЖЛИВО. Для присвоєння треба використовувати масиви або функції користувача.

Наприклад, можливо розрахувати та вивести значення виразу, вбудованої функції чи функції користувача з ранжованою змінною  $x$ :  $2*x$ ,  $\sin(x)$ , – але присвоїти значення цього виразу іншій змінній  $y:=2*x$  – неможливо.

### Перевірка умов

При виконанні розрахунків часто доводиться оцінювати різноманітні умови. Це трапляється при описуванні кусково-неперервних функцій, імпульсних сигналів. Такі дії називаються логічними або умовними. Вони реалізують виконання алгоритму з розгалуженням.

У розгалужених алгоритмах присутні кілька гілок обчислювального процесу. Вибір конкретної гілки залежить від виконання (чи невиконання) заданих умов.

Умовні дії в системі реалізовано у вигляді унітарних логічних операторів (виразів відношень), логічних операцій, логічних виразів. Перевірка умов та розгалуження дій – логічної функції **if**.

Введення символів логічних умов проводиться з клавіатури або кнопками панелі **Boolean** (рис. 3.16).

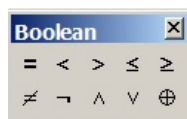


Рис. 3.16. Панель **Boolean**

### Унітарні логічні оператори – вирази відношення

Ці вирази використовуються для порівняння двох арифметичних виразів між собою. Вираз відношення записуються в вигляді:

**<вираз A> <знак відношення> <вираз B>**,

де в якості знака відношення виступають символи  $<$ ,  $>$ ,  $=$ ,  $\leq$ ,  $\geq$ ,  $\neq$ . Якщо задане відношення виконується, то вираз відносин набуває значення рівне 1 («істина»), в іншому випадку – 0 («брехня»).

Знак операції порівняння не є знаком виведення значень обчислень. Знак операції порівняння називається «жирним  $=$ ».



На відміну від мов програмування можна відразу в одному виразі перевіряти кілька умов шляхом додавання знаків відношення і арифметичних виразів:

$$x := 6 \quad 2 \leq x \leq 8 = 1$$

### Логічні операції

Логічна операція «АБО» (OR)

Позначається знаком «+» або «∨» і записується у вигляді

**<Логіч.вир.1> + <Логіч.вир. 2>**

**<Логіч.вир.1> ∨ <Логіч.вир. 2>**

Результат операції дорівнює 0, якщо обидва логічних вирази дорівнюють 0 і дорівнює 1 для всіх інших значень логічних виразів.

Логічна операція «І» (AND) позначається знаком «\*» або «∧» і записується у вигляді

**<Логіч.вир.1>. <Логіч.вир. 2>**

**<Логіч.вир.1> ∧ <Логіч.вир. 2>**

Результат дорівнює 1, якщо обидва логічних вирази дорівнюють 1 і дорівнює 0 для всіх інших значень логічних виразів.

Логічна операція «НІ» (NOT) позначається знаком «¬» і записується у вигляді

**¬ <Логіч.вираз>**

Результат дорівнює зворотним результату логічного виразу.

Логічна операція «ВИКЛЮЧАЮЧЕ АЛЕ» (XOR) позначається знаком «⊗» і записується у вигляді

**<Логіч.вираз 1> ⊗ <Логіч.вираз 2>**

Результат дорівнює 1, якщо тільки одне з логічних вирази дорівнює 1 і дорівнює 0 для всіх інших значень логічних виразів.

## Логічні вирази

*Логічним виразом* називається конструкція, складена з унітарних логічних операторів, логічних операцій, арифметичних операцій. Значення логічного виразу обчислюється зліва направо. Вищий пріоритет мають дії в круглих дужках, далі логічні операції (AND, OR, XOR), найнижчий – вирази відношень (<, >, =).

### Умовна функція **if**

Ця функція записується у наступному вигляді (символи **if** вводяться з клавіатури):

**if** (<умова>, <вираз 1>, <вираз 2>)

В якості умови можна застосовувати арифметичний вираз цілого типу або логічний вираз.

Правило обчислення умовної функції **if**: якщо вираз більше 0, то функція приймає значення рівне значенню виразу 1, якщо умова дорівнює 0, то функція набуває значення рівне значенню виразу 2. В якості виразів може використовуватися будь-яка припустима арифметична, логічна дія і сама функція **if**.

Функція реалізує структуру «якщо – то – інакше», тобто повну альтернативу. При програмуванні алгоритмів з розгалуженням по трьох та більше умовах функція **if** застосовується в якості виразів 1, 2. Така структура називається вкладеними умовними функціями.

При перевірці умов, що описують діапазони значень, перевірку слід проводити послідовно з боку менших або більших значень.

Для підготування виразів з метою перевірки умов в системі існують такі корисні функції, як:

Ceil(x,y) – функція повертає найменше число, яке кратне у та більше x

ceil(x) – функція повертає найменше ціле число, яке більше x

Floor(x,y) – функція повертає найбільше число, яке кратне у та менше x

floor(x) – функція повертає найбільше ціле число, яке менше x

- $\text{Round}(x,y)$  – значення  $x$  округляється до найближчого числа за правилами математики, яке кратне  $y$ .
- $\text{round}(x,n)$  – число  $x$  округляється до  $n$  знаків після десяткової точки. Якщо другий аргумент відсутній, то округлення відбувається до найближчого цілого. Від'ємне значення  $n$  визначає кількість розрядів цілої частини числа
- $\text{Trunc}(x,y)$  – функція повертає добуток від множення  $y$  на цілу частину ділення  $x$  на  $y$ . Якщо  $|x| < |y|$ , то повертається нуль.
- $\text{trunc}(x)$  – функція повертає цілу частину  $x$
- $\text{signum}(x)$  - функція повертає 1, коли  $x=0$ , в інших випадках -  $x/|x|$
- $\text{sign}(x)$  – повертає 0, коли  $x=0$ , 1 – коли  $x>0$ , -1 – коли  $x < 0$
- $\Phi(x)$  – функція Хевісайда (функція сходинки). Повертає 0, коли  $x<0$ , повертає 1, коли  $x>0$
- $\text{max}(c)$  – функція повертає значення максимального елемента списку аргументів. В якості списку може бути масив
- $\text{min}(c)$  – функція повертає значення мінімального елемента списку аргументів. В якості списку може бути масив

### **Робота з векторами та матрицями**

*Масиви* – це об'єднання декількох однотипних об'єктів (чисел, символів, рядків, тощо), які розглядаються як єдине ціле. Доступ до кожного елемента масиву здійснюється за допомогою індексу. Індекс є даним порядкового типу, що служить своєрідним іменем елемента в масиві. При обробці елемента масиву безпосередньо за іменем елемента масиву повинен слідувати індекс. Індекс не повинен виходити за межі, визначені діапазоном. Розрізняють одновимірні масиви, що містять елементи в одному рядку - вектори, двовірні масиви, елементи яких розташовані по рядкам і стовпцям - матриці, багатовимірні масиви складної структури.

В *Mathcad* матрицею  $A$  розміром  $R \times C$  вважається сукупність даних, які розташовані у вигляді таблиці з рядків та стовпців

$$A = \begin{bmatrix} a_{1,1} & \dots & a_{1,C} \\ a_{2,1} & \dots & a_{2,C} \\ a_{R,1} & \dots & a_{R,C} \end{bmatrix}$$

Накоротко матриця позначається через свій елемент. Індеси матриці розташовують після імені нижче назви. Перший індекс визначає рядок, другий – стовпець.

$$A \rightarrow (a_{i,j}) \quad (i=1,2\dots R, \quad j=1,2\dots C)$$

У середовищі *MathCAD* допустимі одномірні масиви-вектори й двомірні масиви - матриці. Вектор є матрицею, в якій кількість стовпців становить 1.

Обробка масивів за вибором користувача здійснюється поелементно або за допомогою матричних (векторних) операцій. Для вибору елемента масиву використовуються *індексовані змінні*. *Індексована змінна Mathcad* аналогічна елементу масиву в мовах програмування. Індеси масивів в можуть приймати тільки цілі значення.

### Введення індексованих змінних

Вбудовані засоби *Mathcad*, які призначені для роботи з матрицями, зібрані на панелі *Matrix* (рис. 3.17).

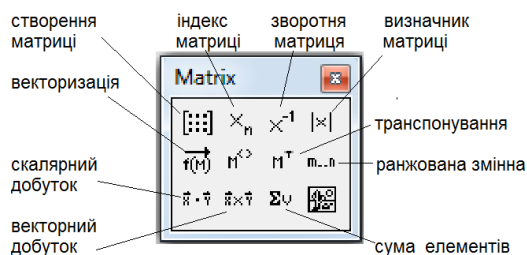


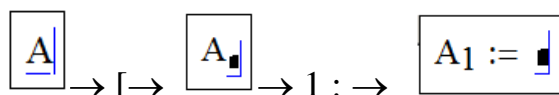
Рис. 3.17. Матрична панель

Початковий індекс матриць та векторів задається системною змінною **ORIGIN**. За замовчанням значення початкового індексу дорівнює 0. В системі індекс масивів може приймати не тільки натуральні, а й цілі значення, тобто бути від'ємним. Змінити значення початкового індексу можна безпосередньо в тексті документу або у вікні системних змінних через пункт меню **Tools- Worksheet Options**.

Індекс змінної виступає ключем при обчисленнях чи реалізації ітераційних алгоритмів.

Визначення матриць можливо двома способами:


1. Безпосереднє присвоювання значення індексованій змінній виконується натисканням клавіші «[» після імені елемента. При цьому на екрані з'являється маркер індексу, на місце якого потрібно ввести необхідне значення



Якщо визначається елемент  $a_{i,j}$ , то автоматично створюється відповідний масив розміром  $i \times j$

$$x_2 := 10 \qquad A_{2,3} := 30$$

$$x = \begin{pmatrix} 0 \\ 0 \\ 10 \end{pmatrix} \qquad A = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 30 \end{pmatrix}$$

2. Визначення вектора (матриці) в діалоговому вікні. Вікно активується клавішами *Ctrl+M* клавіатури, пунктом меню **Insert-Matrix**, кнопкою  панелі. У вікні потрібно визначити кількість рядків (*rows*) і стовпців (*columns*) матриці.

Після введення значень у полі документа на місці знаходження курсора відображається структура матриці, яка створюється. На місці маркерів необхідно ввести потрібні значення.

### Операції з матрицями

Над векторами та матрицями можна проводити обчислення, як і над числами. Треба пам'ятати, що результат арифметичної дії над масивами залежить від типу операндів. Дія з скаляром-числом та масивом проводиться над кожним елементом масиву.

Дії над двома масивами виконуються за правилами матричної математики!

Основні матричні функції наведені в табл. 3.5.

Таблиця 3.5. Основні матричні операції *MathCAD*

Оператор	Дія
length(v)	повертає число елементів вектора v
last(v)	повертає індекс останнього елемента вектора v
rows(M)	повертає число рядків матриці M
cols(M)	повертає число стовпців матриці M
M <sup>&lt;i&gt;</sup>	виділяє i-й стовпець матриці M
identity (n)	формує одиничну квадратну матрицю розміром n x n
diag (v)	формує діагональну матрицю, на головній діагоналі котрої розташовані елементи вектора v.
matrix (m,nf)	формує матрицю, в якій (i,j)-й елемент дорівнює значенню функції f(i,j)
augment (A, B)	формує матрицю приєднанням горизонтально матриць A, B

Таблиця 3.5. Продовження

stack (A, B)	формує матрицю приєднанням після останнього рядка матриці A матрицю B вниз
submatrix (A, ir, jr, ic, jc)	вирізає з матриці A підматрицю з рядками з ir по jr та стовпцями з ic по jc
rank(A)	ранг матриці - кількість лінійно незалежних рядків та стовпців
tr(A)	слід – сума діагональних елементів
mean(A)	середнє значення елементів масиву
norme(M)	евклідова норма матриці $\sqrt{\sum_{i=1}^n \sum_{j=1}^m M_{i,j}^2}$
sort (v)	сортує елементів вектору за зростанням їх значень
reverse (v)	переставляє елементів в зворотному порядку
csort (M,n)	переставляє рядки матриці так, що n-й стовпець сортується за зростанням
rsort (M,n)	переставляє стовпці матриці так, що n-й рядок сортується за зростанням
A <sup>-1</sup>	зворотня матриця

$ v ,  A $	довжина вектора $v$ , детермінант матриці $A$
$A^T$	транспонування $A$ (Ctrl+1)
$A^*$	комплексне спряжіння ( “ )
$\sum v$	сума елементів вектора $v$ (Ctrl+4)

### Векторизація обчислень

Будь-яке обчислення, яке *MathCAD* може виконати з поодинокими значеннями, він може виконувати поелементно з векторами і матрицями (табл. 3.6). Це можна реалізувати двома способами: послідовно виконуючи дії над кожним елементом масиву або використанням *оператора векторизації*. Для введення цього оператора необхідно над виділеним об'єктом натиснути одночасно клавіші *Ctrl+-* (обрати символ  $\rightarrow$  з панелі *Matrix*). Векторизований об'єкт зображується з стрілкою нагорі.

Оператор векторизації змінює зміст операцій. Наприклад, якщо  $A$  – це деяка матриця. Тоді запис **exp(A)** є некоректним, оскільки аргументом функції **exp** повинна бути проста змінна, а не матриця. Застосування до цієї функції оператора векторизації призводить до обчислення функції **exp** від кожного елемента матриці і результатом також є матриця.

Таблиця 3.6. Оператори арифметичних дій з масивами

Дія	Вигляд	Опис
Множення	$A * k$	Множить кожен елемент $A$ на скаляр $k$
	$w * v$	Скалярний добуток вектора-рядка на вектор-стовпець
	$A * B$	Матричне множення: рядок – стовпець
	$A * v$	Множення матриця – вектор-стовпець
Ділення	$A / k$	Ділить кожен елемент масиву на $k$
Складання	$A + B$	Поелементне складання масивів
	$A + k$	Додавання до кожного елемента $A$ числа $k$

Віднімання	A – B	Поелементне віднімання масивів
	A-k	Віднімання з кожного елемента A числа k
Векторне множення	u x v	Добуток векторів [Ctrl+8]. Результат - вектор

Позначення: матриця – A, B, M, вектор – v, u, скаляр - k.

### Виведення результатів

В системі передбачено відображення масивів в формі матриці та в формі таблиці.

$$b = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} +$$

	0	1	2	3
0	1	0	0	0
1	0	1	0	0
2	0	0	1	0
3	0	0	0	1

стиль «матриця»

стиль «таблиця»

Визначення стилю відображення масивів проводиться пунктом **Matrix display style** на сторінці меню **Format-Result Display Options**. В списку знаходяться три стилі **Automatic**, **Matrix**, **Table** (рис. 3.18).

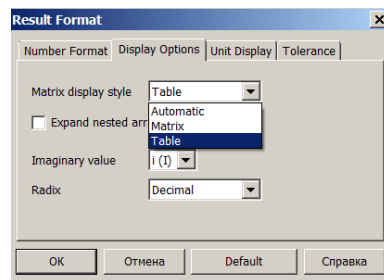


Рис. 3.18. Вікно визначення стилю масивів

### Стиль Matrix (Матриця)

Матриця завжди відображується як прийнято з точки зору математики. При цьому видимі всі елементи масиву. Однак, коли масив має великі розміри, то зображення займає дуже багато місця на екрані. Крім того, важко проводити ідентифікацію конкретного елемента масиву, бо номери рядків та стовпців на екрані не відображаються.

### Стиль Table (Таблиця)

Матриця завжди відображується як таблиця з виведенням у перший рядок номерів стовпців, в перший стовпець – номери рядків. На екран



виводиться тільки частина масиву в діапазоні індексів 1 ... 10. Перегляд частини масиву, що залишилася, проводиться за допомогою вертикального та горизонтального повзунків.

Значення максимального індексу елемента для відображення можна відредагувати в пункті **Properties** контекстного меню масиву на сторінці **Data Range**.

Зображення у вигляді таблиці допускає вирівнювання матриці відносно знаку прирівнювання операцією **Alignment** контекстного меню масиву.

Зображення індексів в таблиці можна відмінити. Для цього в пункті **Properties** контекстному меню масиву на сторінці **Display** слід змінити поле **Show column/row labels**.

### Стиль Automatic

Система сама визначає стиль виведення. Для розмірів масиву менше 10 використовується стиль матриця, для більших – таблиця. За замовчанням встановлено стиль **Automatic**.

## 3.1.2. Приклади та завдання для самостійного виконання

*Приклад 3.1.1.* Створити в *MathCAD* сторінку (рис. 3.19). Малюнок вставити з папки *MathCAD template*. Результат зберегти в *web* форматі.

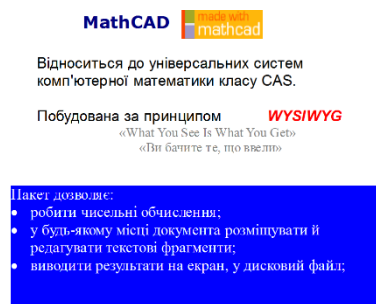


Рис. 3.19. Макет документа прикладу 3.1.1

### РОЗВ'ЯЗАННЯ

Сторінку можна створити з 5-ти до 2-х текстових областей. Оберемо 3. Перша область має шрифт *Verdana* 16 темно-синього кольору. Друга – шрифти *Arial* 14 чорного кольору, *Arial bold italic* 16 червоного кольору, *Times New Roman italic* 14 сірого кольору. В ній розташовані два абзаци.

Вирівнювання другого абзацу – по центру. Третя область має шрифт *Times New Roman* 14 білого кольору та синій фон. В ній використано список.

Форматувати текстову область можна перед її створенням за рахунок визначення стилів документа або вже після введення тексту. Покажемо обидва способи.

1. Обрати пункт меню **Format – Style**.

2. Обрати **Text styles – Heading1 – Modify – Font**. У вікні визначити: *Font – Verdana*, *Font size – 16*, *Font style – полужирный*, *Color – темно синий* (рис. 3.20).

3. Обрати **Text style – List – Font**. У вікні визначити: *Font – Times New Roman*, *Font size – 14*, *Font style – нормальный*, *Color – белый*.

4. Обрати **Text style – Normal – Font**. У вікні визначити: *Font – Arial*, *Font size – 14*, *Font style – нормальный*.

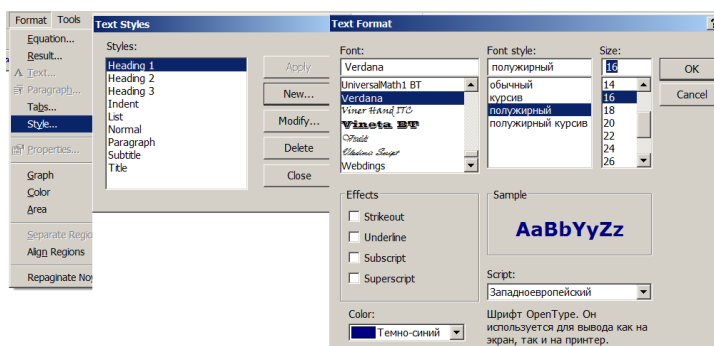


Рис. 3.20. Визначення стилю *Heading1*

5. Встановити курсор на місце першого рядка. На панелі **Formatting** обрати стиль *Heading1*. Ввести «MathCAD».

6. Відкрити графічною програмою зображення з папки «*./MathCADxx/template/madewithmathcad.gif*», скопіювати його в *Clipboard* та вставити документ *MathCAD*. Відкрити через контекстне меню сторінку *Protect* властивостей та відмінити блокування редагування (*Protect region from editing*). Розташувати малюнок в потрібне місце та надати йому потрібного розміру (рис. 3.21).

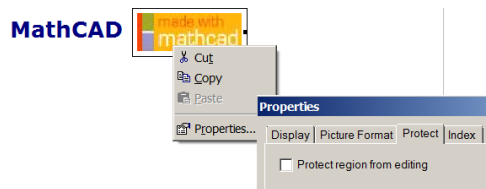


Рис. 3.21. Перший рядок документа

7. Встановити курсор на місце другого рядка. На панелі **Formatting** обрати стиль *Normal*.

8. Ввести наступний текст (рис. 3.22):

«Відноситься до універсальних систем  
комп'ютерної математики класу CAS.

Побудована за принципом **WYSIWYG**

«What You See Is What You Get» «Ви бачите те, що ввели»».

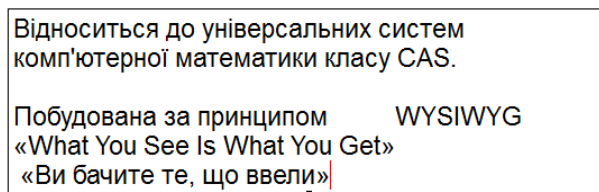


Рис. 3.22. Друга область

9. Мишею виділити **WYSIWYG**. В контекстному меню обрати пункт *Font*. Визначити стиль шрифту *полужирный курсив*, колір – *красный*.

10. Мишею виділити останні два рядка. В контекстному меню обрати пункт *Font*. Визначити шрифт *Times New Roman*, стиль шрифту – *полужирный*, колір – *темно-серый*.

11. На панелі **Formatting** обрати тип вирівнювання – *по центру* (рис. 3.23).

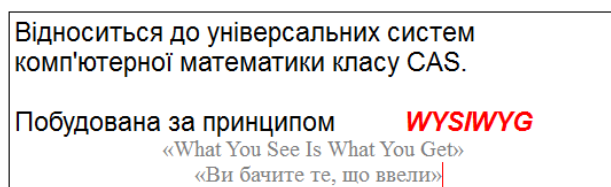


Рис. 3.23. Відформатована друга область

12. Встановити курсор на місце списку. На панелі **Formatting** обрати стиль *List*. Створити текстову область. В контекстному меню **Properties**

**-Display - Background** позначити поле *Highlight region*, кнопкою *Choose color* обрати колір – *blue*. Ввести текст «Пакет дозволяє:» (рис. 3.24).



Рис. 3.24. Третя область

13. На панелі **Formatting** включити кнопку створення немаркованого списку. Ввести наступний текст:

«робити чисельні обчислення;

у будь-якому місці документа розміщувати й редагувати текстові фрагменти;

виводити результати на екран, у дисковий файл.»

14. Зберегти документ як *Web page (MathCAD)*.

*Завдання 3.1.1.* Текстові області.

1. Створити документ.

В документі горизонтально поруч розмістити дві текстові області:

- Перша область повинна містити текст цього завдання. Колір шрифту – чорний. Перший рядок – *Arial 16 bold*. Заголовок списку – *Arial 14*, список – *Arial 14 italic*.
- Друга область: 1-й рядок «Виконав студент(ка) ПІБ» . Другий рядок «Група ХХ-ХХ дата». Колір – синій. Шрифт тексту – *Courier 14*, «ПІБ та номер групи, дата» шрифт – *Courier 14 bold*. Вирівнюванн – по правому краю.

2. Документ зберегти в форматі *Word*.

*Приклад 3.1.2.* Одиночні розрахунки.

Для значення  $x=2$  обрахувати вираз з функцією Беселя першого роду  $J_0(x)$  та гіпергеометричною функцією  $\Gamma(x)$ :

$$J_0 + \frac{\Gamma(x)}{\sin(x)} e^{2-5i}.$$

РОЗВ'ЯЗАННЯ

**MathCAD**

$x:=2$  – визначаємо змінну-аргумент виразу

Функції можна вводити з клавіатури, з меню **Insert-Function** або кнопкою **f(x)** панелі **Standard**.

Натиснути кнопку **f(x)** панелі **Standard** → обрати зі списку групи *Bessel* функцію Беселя першого роду →  $J_0(x)$  → вставити на місце маркера змінну **x**, додати дію →  $J_0(x) +$  → в меню **Insert-Function** з групи *Special* обрати функцію *Gamma* →  $J_0(x) + \Gamma(\cdot, \cdot)$  → видалити другий маркер, на місце першого маркера поставити аргумент **x** → завершити вираз.

$$J_0(x) + \frac{\Gamma(x)}{\sin(x)} \cdot e^{2-5i} = 2.529 + 7.792i$$

### MathCAD Prime

$x:=2$  – визначаємо змінну-аргумент виразу

Функції вводяться зі сторінки **Function** стрічки.

Обрати функцію Беселя першого роду  $J_0$  групи **Bessel** стрічки → вставити на місце маркера змінну **x**, додати дію →  $J_0(x) +$  → Обрати функцію гамма з меню **All Functions** стрічки →  $J_0(x) + \Gamma(\cdot)$  → на місце маркера поставити аргумент **x** → завершити вираз.

$$J_0(x) + \frac{\Gamma(x)}{\sin(x)} \cdot e^{2-5i} = 2.529 + 7.792i$$

#### Приклад 3.1.3. Системи числення (MathCAD).

Розрахувати суму. Результат вивести в десятковій, бінарній, восьмеричній, шістнадцятиричній системах. Доданки;  $1110_2$ ,  $21_8$ ,  $3F_{16}$

#### РОЗВ'ЯЗАННЯ

Звичайно, можна перевести доданки у відповідну систему та скласти. Для десяткової системи це буде:

$$1110_2=14 \quad 21_8=17 \quad 3F_{16}=63 \quad 14+17+63=94.$$

Робити це немає потреби. Записати суму та отримати результат за замовчанням в десятковій системі можна простим додаванням уточнюючого символу системи числення після числового значення:

$$1110b+21o+3Fh= 94.$$

Обрання в меню пункту **Format-Result-Display options-Radix-Octal** переводить результати в документі в восьмеричну систему:

$$1110b+21o+3Fh= 136o.$$

Обрання в меню пункту **Format-Result-Display options-Radix-Hexadecimal** переводить результати в документі в шістнадцятиричну систему:

$$1110b+21o+3Fh= 5eh.$$

Обрання в меню пункту **Format-Result-Display options-Radix- Binary** переводить результати в документі в двійкову систему:

$$1110b+21o+3Fh= 1.1bx10b110b.$$

Примітка. Результат видано в двійковій системі, але вигляд його незручний. Причиною цього є те, що за замовчанням виставлено формат результату порогом переходу в експоненційну форму 3 цифри. В результаті їх більше, тому форма – експоненційна.

Для корекції слід обрати в меню **Format-Result-Number Format -Exponential Threshold** значення 8 та отримати наступний результат

$$1110b+21o+3Fh= 1011110b.$$

*Питання.* Як перевести в потрібну систему не всі результати в документі, а локальний вираз??

*Приклад 3.1.4.* Розрахувати вираз згідно наведеної формули

$$h(r) = \left[ \frac{2I_1\left(\frac{2\pi D_{зр} r}{\lambda f}\right)}{2\pi D_{зр}} \right]^2$$

Розрахунок провести для  $\lambda=5.5$  мкм,  $D_{зр}=0.1$  мм,  $f=25$  мм. Діапазон зміни аргументу  $r$  від 0 мм до 1 мм з кроком 0.1 мм.

$$D := 0.1 \quad \lambda := 0.55 \cdot 10^{-2} \quad f := 25$$

РОЗВ'ЯЗАННЯ

$$x := \frac{2 \cdot \pi \cdot D}{\lambda \cdot f}$$

– задаємо вихідні дані. Дані масштабовано так, щоб всі вони

$$h(r) := \left( 2 \cdot \frac{J_1(x \cdot r)}{x \cdot r} \right)^2$$

$$r := 10^{-6}, 0.1..1$$

становили значення в міліметрах ( $1\text{мкм}=10^{-3}\text{ мм}$ )

– уводимо проміжну змінну

– задаємо функцію користувача

– уводимо ранжовану змінну  $r$  як аргумент для розрахунку з першим значенням 0.000001, кроком 0.1, останнім значенням 1. Перше значення не може дорівнювати 0, бо результатом розрахунку цього значення аргумент функції має невизначеність  $0/0$ .

– виводимо таблиці результату. Перший стовпчик є вектором із десятих значень аргументу  $r$ , другий стовпчик є вектором із десятих значень результату  $h(r)$ .

$r =$	$h(r) =$
0	0
0.1	0.949
0.2	0.808
0.3	0.613
0.4	0.405
0.5	0.226
0.6	0.098
0.7	0.027
0.8	$1.562 \cdot 10^{-3}$
0.9	$2.751 \cdot 10^{-3}$
1	0.012

### Завдання 3.1.1.

а) Для комплексних змінних  $6+3i$  та  $9.1e^{-i0.33}$  визначити дійсну, уявну частини, модуль, фазу.

б) Обчислити значення  $z_4$ , записати результат в алгебраїчній та експоненціальній формах.

$z_1$	$z_2$	$z_3$		$z_4$
$4+3i$	$1.82 \cdot e^{-1.2i}$	$\sqrt{3}-2i$	$z_4 = \frac{z_1^2 \cdot z_2}{z_3} + z_3$	$11.8182 + 11.9291i$

**Завдання 3.1.2.** Обчислити вирази. Порівняти отримані результати з наведеними, визначити формат результату.

$$\text{а) } \frac{23.1 \cdot \sqrt{3\pi - \frac{\ln(54)}{16}}}{5 - \pi^{41}} = -81.9085 \times 10^{-12}$$

$$\text{б) } e^{-2.5} \ln(11.3)^{0.3} - \sqrt{\frac{\sin(2.45\pi) + \cos(3.78\pi)}{\operatorname{tg}(3.3)}} = -3.21$$

$$\text{в) } \frac{2,625 - \frac{2}{3} \cdot 2 \cdot \frac{5}{14}}{(3\frac{1}{12} + 4,375) \div 19\frac{8}{9}} = 2.80952 \times 10^0$$

$$\text{г) } \frac{\cos(\frac{2\pi}{6}) \sin(\frac{\pi}{3})}{\operatorname{tg}(\frac{2\pi}{9}) - \lg(44)} = -0.538$$

**Завдання 3.1.3.** Для визначених значень змінних обчислити вирази. Порівняти отримані результати з наведеними, визначити формат результату.

$$\text{а) } \frac{a^3 \sqrt{x} - \sin(x+a)}{e - |x-a|} = -0.149 \quad \text{для } x=0.5 \quad a=0.7$$

$$\text{б) } \frac{3\sqrt{a} \cdot \ln|a|}{\sin(e^{2x}) \cos(xa^3)} = -2.212 \quad \text{для } x=0.5 \quad a=0.7$$

$$\text{в) } \frac{\sin(\frac{\lambda\pi}{D})}{\frac{\lambda\pi}{D}} = 0.999 \quad \text{для } \lambda=0.9 \quad D=50$$

Примітка: функція має значення 1.0 тільки для аргументу 0.

$$\text{г) } \sqrt{\frac{4}{e^x} - \operatorname{cth}^3(x) \cos(|x \sin(x^2) - \ln(x)|)} = -0.559 \quad \text{для } x=3$$

$$\text{д) } \frac{b^3 + \sin^2(a)}{\arccos(b^2) + e^{\frac{b}{2}}} = 239/605 \quad \text{для } a=83^0 \quad b=0.12$$

**Завдання 3.1.4.** Застосовуючи функції користувача для спрощення записів, обрахувати вираз для значень  $x=0.1$ ,  $y=1.2$ .



$$\text{a) } \operatorname{arctg}\left(\frac{\sqrt[3]{x - \sin(y)}}{\sqrt{1-x^2}} - \frac{|x|\sqrt{1-x^2}}{\sqrt[3]{x - \sin(y)}}\right)$$

$$\text{б) } \frac{(\sin(y) + \sin(2y) + \sin(3y))^4}{1 + \frac{\sin(y) + \sin(2y) + \sin(3y)}{1 + e^x}} + \sqrt{\frac{1 + \sin(y) + \sin(2y) + \sin(3y)}{1 + e^x}}$$

$$\text{в) } \frac{\sqrt{\cos(2y) + \sin(4y) + \sqrt{e^x + e^{-x}}}}{(e^x + e^{-x})^3 (\cos(2x) + \sin(4y) - 2)^2} \quad \text{г) } \frac{1 + \arcsin(\cos(2y))}{2^x + 3^{-x}} + \left(\frac{2^x + 3^{-x} - 1}{1 + \arcsin(\cos(2y))}\right)^2$$

*Завдання 3.1.5.* Обрахувати вираз з застосуванням розширених арифметичних операторів

$$\text{а) } \prod_{i=1}^{\infty} \left(1 - \frac{1}{i^2}\right) \quad \text{б) } \sum_{k=1}^{10} \frac{1}{k^4} \quad \text{в) } \sum_{i=3}^{20} e^{-\frac{i}{2}}$$

*Завдання 3.1.6.* Розрахувати значення функції в заданому діапазоні

$$\text{а) } \frac{\sin^2(x)}{1 + \cos(x)} + e^{-x} \ln(x) \quad \text{діапазон : } x = 0.2 \dots 2.4 \text{ з кроком } 0.2$$

$$\text{б) } e^{-x} \sin(10x) \quad \text{діапазон : } x = 0 \dots 1 \text{ з кроком } 0.1$$

$$\text{в) } \cos(x-y) + \sin^2(x+y) \quad \text{діапазон: } x = 0.1 \dots 1, \text{ крок } 0.4, y = 1 \dots 2 \text{ крок } 0.3.$$

*Завдання 3.1.7.* Модифікувати функцію користувача з прикладу 3.1.4 так, щоб вона мала додатково за аргумент  $f$ . Розрахувати значення для  $f = (25, 50, 100)$  мм,  $\lambda = 0.55$  мкм,  $D_{\text{зр}} = 1$  мм. Діапазон зміни аргументу  $r = (0..0.1)$  мм з кроком 0.01 мм. Таблиці значення аргументу  $r$  та результати для  $f = (25, 50, 100)$  мм розташувати в одному рядку.

*Приклад 3.1.5.* Описати прямокутну функцію

$$\operatorname{rect}(x, a) = \begin{cases} 0, & |x| > \frac{a}{2} \\ 1, & |x| \leq \frac{a}{2} \end{cases}$$

### РОЗВ'ЯЗАННЯ

$$\operatorname{rect}(x, a) := \left(\Phi\left(x + \frac{a}{2}\right) - \Phi\left(\frac{a}{2} - x\right)\right) \quad \text{з використанням функції Хевісайда}$$

$$\operatorname{rect}(x, a) := \operatorname{if}\left(-\frac{a}{2} < x < \frac{a}{2}, 1, 0\right) \quad \text{з використанням функції if}$$

$rect(x, a) := if(|x| > \frac{a}{2}, 0, 1)$  функцією **if** зі скороченою умовою

$rect(x, a) := (|x| \leq \frac{a}{2})$  з використанням унітарного логічного оператора

*Приклад 3.1.6.* Описати функцію  $y(x)$

$$y(x) = \begin{cases} \frac{\sqrt{x}}{2} & x \geq 1 \\ \frac{\sqrt[3]{x}}{3} & 0 < x < 1 \\ \frac{\sqrt[4]{|x|}}{4} & x \leq 0 \end{cases}$$

**РОЗВ'ЯЗАННЯ**

$$y(x) := if(x \geq 1, \frac{\sqrt{x}}{2}, if(x > 0, \frac{\sqrt[3]{x}}{3}, \frac{\sqrt[4]{|x|}}{4}))$$

*Приклад 3.1.7.* Описати функцію (рис. 3.25) із застосуванням логічної функції та функції Хевісайда.

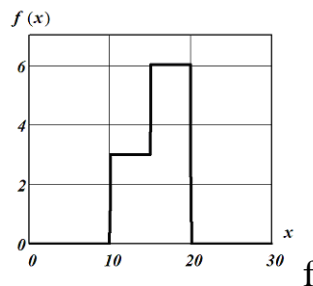


Рис. 3.25. Графік функції з прикладу 3.1.7

**РОЗВ'ЯЗАННЯ**

$f(x) := if(x < 10, 0, if(x < 15, 3, if(x < 20, 6, 0)))$  – логічна функція

$f(x) := 3(\Phi(x-10) + \Phi(x-15) - 2\Phi(x-20))$  – функція Хевісайда

*Приклад 3.1.8.* Описати функцію, яка повертає значення 1, якщо точка з координатами  $x, y$  знаходиться всередині фігури (рис. 3.26), 0 – якщо ні.

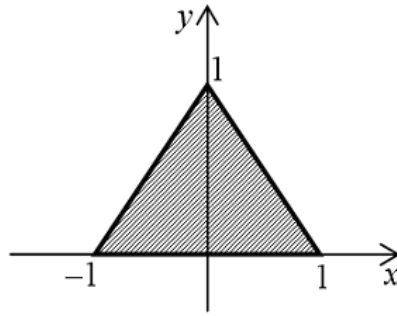


Рис. 3.26. Область з прикладу 3.1.8

### РОЗВ'ЯЗАННЯ

Область обмежена прямими  $y=0$ ,  $y=x+1$ ,  $y=1-x$  та визначається умовами  $y>0$  та  $y<x+1$  та  $y<1-x$  одночасно.

$$z(x,y) := (y > 0) \wedge (y < x + 1) \wedge (y < 1 - x) \text{ або}$$

$$z(x,y) := \text{if}((y > 0) \wedge (y < x + 1) \wedge (y < 1 - x), 1, 0)$$

*Завдання 3.1.8.* Описати функцію одиничної сходишки.

*Завдання 3.1.9.* Описати функцію (рис. 3.27) із застосуванням логічної функції та вбудованої функції Хевісайда.

*Завдання 3.1.10.* Описати функцію, яка повертає значення 1, якщо точка з координатами  $x, y$  знаходиться всередині фігури (рис. 3.28), та 0 - якщо ні.

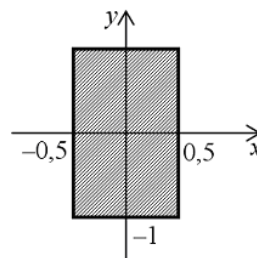
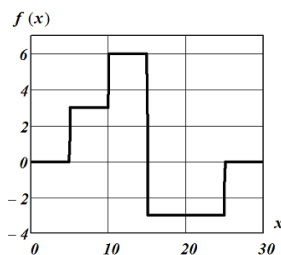


Рис. 3.27. Функція до завдання 3.1.9 Рис. 3.28. Область до завдання 3.1.10

*Завдання 3.1.11.* Описати функцію

a) 
$$f(x,y) = \begin{cases} \sin(x + y^2) - e^{2x} & xy < 2 \\ \ln(x + \sqrt[3]{|y|}) & 2 < xy < 3 \\ \cos(x + y^3) + e & xy > 3 \end{cases}$$

$$b) f(x,y) = \begin{cases} \min\left(\frac{\cos(x^2)}{y}, \sqrt[5]{y} \sin\left(\frac{x}{y}\right)\right) & x < y \\ \max\left(\sqrt{|\sin(xy)|}, \cos\frac{\sqrt{x}}{y+1}\right) & x \geq y \end{cases}$$

Приклад 3.1.9. Створити вектор значень функції  $e^{-x} \sin(10x)$  в діапазоні  $x=0 \dots 1$  з кроком 0.1.

### РОЗВ'ЯЗАННЯ

На визначеному діапазоні функція має  $N = \frac{X_{\max} - X_{\min}}{\text{step}} + 1 = \frac{1-0}{0.1} + 1 = 11$

відліків.

$$x_{\text{start}} := 0 \quad x_{\text{stop}} := 1 \quad \text{step} := 0.1$$

Визначаємо вихідні дані

$$N := \frac{x_{\text{stop}} - x_{\text{start}}}{\text{step}} + 1$$

Визначаємо довжину вектору

$$i := 1..N$$

Визначаємо ранжовану змінну

$$x_i := x_{\text{start}} + (i - 1) \cdot \text{step}$$

Вводимо проміжний вектор для того, щоб мати змогу присвоїти значення новій змінній - елементу вектора

$$y_i := e^{-x_i} \cdot \sin(10 \cdot x_i)$$

	1
1	0
2	0.761
3	0.744
4	0.105
5	-0.507
6	-0.582
7	-0.153
8	0.326
9	0.445
10	0.168
11	-0.2

Примітка. Ефективніше можна створити вектор-результат з використанням операції векторизації

$$z := \overrightarrow{\left( e^{-x} \cdot \sin(10 \cdot x) \right)}$$

Приклад 3.1.10. Знайти суму додатних елементів матриці A розміром 3x3.

### РОЗВ'ЯЗАННЯ

1. За допомогою вікна створення матриць визначаємо матрицю A

A: → Ctrl+M → 3 3 →  $A := \begin{pmatrix} \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \end{pmatrix}$  → на місце маркерів .

вводимо значення →  $A := \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 1 \end{pmatrix}$

2. Для розрахування суми елементів застосуємо розширений оператор підсумування. Треба перевірити всі елементи матриці, тому використовуються два оператори. Межі в операторах підсумування визначаємо по першому та останньому індексам матриці A. Суму будемо накопичувати в змінній S. Перевірку додатності елементів матриці A реалізує умовна функція **if**.

S: → Ctrl+Shift+\$ → Ctrl+Shift+\$ →  $s := \sum_{i=1}^{\cdot} \sum_{j=1}^{\cdot} \cdot$  →

$$\rightarrow s := \sum_{i=1}^3 \sum_{j=1}^3 (\text{if}(A_{i,j} > 0, A_{i,j}, 0)) = 37$$

Приклад 3.1.11. Для матриці  $A := \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 1 \end{pmatrix}$  визначити основні параметри.

### РОЗВ'ЯЗАННЯ

Основними параметрами матрицями слугують

детермінант  $|A| = 24$

слід  $\text{tr}(A) = 7$

ранг  $\text{rank}(A) = 3$

$\text{norme}(A) = 14.318$

норма

Завдання 3.1.12. Визначити вектор довжиною 6  $x_i = \frac{1}{i^2 + 1}$ .

а) Переставити елементи в зворотному порядку вбудованою функцією та за допомогою поелементного звертання.

б) Виділити в новий вектор елементи з парними номерами.

в) Замінити всі мінімальні елементи на максимальні.

Завдання 3.1.13. Визначити матрицю  $B = \begin{bmatrix} 11 & -11 & 3 \\ 2 & -2 & 7 \\ 9 & 3 & 3 \end{bmatrix}$ .

а) Обрахувати детермінант та зворотну матрицю.

б) Розрахувати параметр  $r = \sqrt{\sum_{i,k=1}^N a_{i,k}^2}$

Завдання 3.1.14. Обчислити вираз  $2 \cdot A \cdot B - 3 \cdot C \cdot D$ , де:

$$A = \begin{bmatrix} 2 & 3 & -1 & -2 & 4 \\ 10 & 3 & -2 & -1 & 3 \\ 5 & 1 & 1 & -4 & 1 \end{bmatrix} \quad B = \begin{bmatrix} 0 & -1 \\ 1 & -2 \\ -1 & 0 \\ 1 & 2 \\ -2 & 3 \end{bmatrix} \quad C = \begin{bmatrix} 1 & -2 & 1 & 4 \\ -3 & 1 & 0 & 5 \\ 2 & 0 & 3 & -1 \end{bmatrix} \quad D = \begin{bmatrix} 1 & -1 \\ -2 & 0 \\ 3 & -1 \\ 2 & -1 \end{bmatrix}$$

Завдання 3.1.15. З перших двох рядків матриці D (завдання 3.1.14) сформувати матрицю R, а з двох останніх - матрицю T. Перевірити, чи є матриці перестановочними, тобто, чи виконується умова  $|R \cdot R^T - T \cdot T^T| < 10^{-10}$ .

## Практикум 3.2. Програмування

### 3.2.1. Теоретичні положення

Реалізувати алгоритм обчислення в пакеті *MathCAD* можна двома способами:

- вставляючи відповідні оператори чи функції в текст документа *MathCAD*. Такий спосіб називається програмуванням в тексті документа. Він є прийнятним для простих алгоритмів;
- використовуючи підпрограми-функції, які містять конструкції, багато в чому подібні до конструкцій таких мов як PASCAL або C.

В обох випадках реалізуються дії лінійного обчислення, розгалуження, циклічних обчислень.

Програмування із застосуванням підпрограм-функцій реалізує концепцію модульного програмування, коли незалежні алгоритми створюються у вигляді окремих програмних модулів.

Застосування підпрограм-функцій подібне до застосування функцій користувача та також включає два етапи: *опис підпрограми-функції*; *виклик підпрограми-функції*.

### **Опис підпрограми-функції**

Перед тим як використовувати підпрограму-функцію, потрібно її задати, тобто *виконати опис*. Опис підпрограми-функції виконується за тими ж правилами, що й опис функції користувача. Відмінністю є те, що список формальних параметрів може бути відсутнім.

Тіло підпрограми-функції включає будь-яке число локальних операторів присвоювання, умовних операторів і операторів циклу, а також викликів інших підпрограм-функцій і функцій користувача. Для введення операторів тіла підпрограми-функції використовується панель **Programming** (рис. 3.27). В ній розміщено кнопки введення додаткового рядка підпрограми **Add line**, локального оператора присвоювання, умовного оператора **if** та оператора повного вибору **otherwise**, передумовленого циклу **for**, ітераційного циклу **while**, спеціальні оператори переривання обчислень **break**, **continue**, **return**, **on error**.

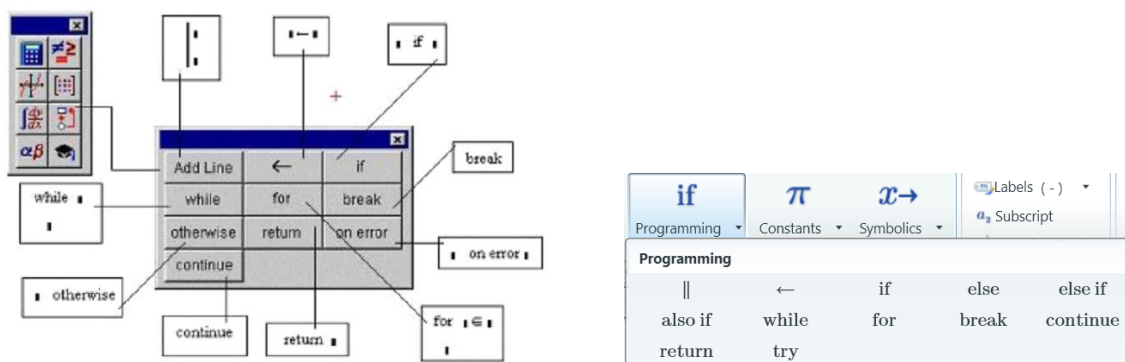


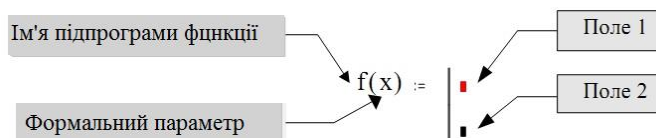
Рис. 3.27. Панель програмування

Коли підпрограма-функція не має формальних параметрів, тоді дані передаються через імена змінних сеансу, визначених вище опису програми-функції. **ВИКОРИСТАННЯ ПІДПРОГРАМ-ФУНКЦІЙ БЕЗ ФОРМАЛЬНИХ ПАРАМЕТРІВ НЕ РЕКОМЕНДУЄТЬСЯ.**

### Порядок опису підпрограми-функції MathCAD

Для введення в робочий документ опису програми-функції необхідно виконати наступні дії:

1. Ввести ім'я підпрограми-функції і список формальних параметрів, взятий у круглі дужки;
2. Ввести символ присвоювання;
3. Додати вертикальну лінію зліва від знака присвоювання клавішею «]» або кнопкою **Add line** панелі **Programming**. На екрані з'явиться вертикальна риска і вертикальний стовпець з двома маркерами для введення операторів, що утворюють тіло програми-функції



Поля шаблону призначені для розміщення в них відповідних операторів алгоритму. Додавання нових рядків проводиться дією **Add line** клавішами клавіатури або кнопкою панелі **Programming**. При цьому нове поле введення додається внизу виділеного на цей момент оператора. Для видалення поля потрібно помістити на нього покажчик і натиснути клавішу



*Delete*, для видалення оператора, потрібно виділити його і натиснути клавішу *Delete*.

Підпрограма-функція повертає результат останньої виконаної дії. Для усунення неоднозначностей рекомендується в самому нижньому полі розміщувати ім'я результату. Результатом може бути число, вектор, матриця, тобто будь-який тип даних пакета.

У наведеному нижче прикладі формальним параметром є проста змінна  $x$ . Підпрограма включає два локальних оператора присвоювання і змінну  $z$ , яка повертається результатом виконання програми-функції.

$$f(x) := \begin{cases} x \leftarrow x + 2 \\ \\ z \leftarrow x^{\frac{1}{3}} \\ \\ z \end{cases}$$

*Локальний оператор присвоювання* використовується для завдання всередині підпрограми значення будь-якої змінної:

**<ім'я змінної> ← <вираз>**

Увага! Використання «звичайного» оператора присвоювання в тілі програми-функції призводить до синтаксичної помилки.

### **Звернення до програми-функції MathCAD**

Для виконання підпрограми-функції необхідно звернутися до імені підпрограми-функції з вказівкою списку фактичних параметрів (якщо в описі підпрограми присутній список формальних параметрів).

Співвідношення між фактичними і формальними параметрами для підпрограми-функції таке ж, як для функцій користувача.

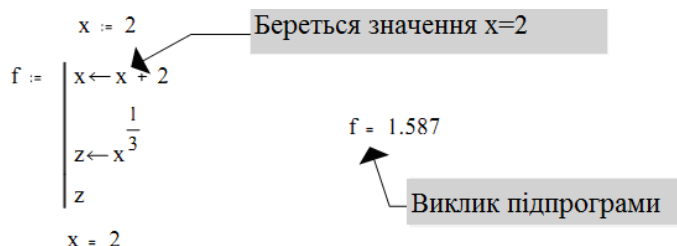
Звернення до підпрограми-функції має відбуватися після опису підпрограми-функції.

Наприклад, звернення до програми  $f(x)$  може мати наступний вигляд:

$$\begin{aligned} x &:= 2 & f(x) &= 1.587 & f(-3.23) &= 0.536 + 0.928i \\ z &:= f(x + 4.5) & z &= 2.041 & & \blacksquare \end{aligned}$$

Передати дані всередину підпрограми-функції можна використанням всередині підпрограми глобальних змінних, які визначені в поточному документі *MathCAD* до опису підпрограми-функції.

Наприклад,



Хоча значення змінної «x» змінилося всередині підпрограми-функції, поза описом підпрограми-функції ця змінна зберігає своє колишнє значення, тому що використовувався локальний оператор присвоєння.

## Різновиди алгоритмів в підпрограмах-функціях

### Лінійні алгоритми

Під *лінійним* алгоритмом розуміється процес, в якому необхідні операції виконуються суворо послідовно. Оператори, що реалізують цей алгоритм у тілі підпрограми-функції також розміщуються послідовно і виконуються всі, починаючи з першого оператора і закінчуючи останнім [9].

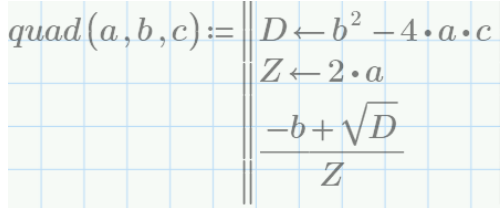
Наприклад. Для створення підпрограми-функції обчислення кореня квадратного рівняння  $ax^2 + bx + c = 0$  за формулою

$$x = \frac{-b + \sqrt{b^2 - 4ac}}{2a} \quad \text{потрібно:}$$

1. Ввести назву підпрограми *quad(a,b,c)*, поставити знак прирівнювання «:=», створити заготовку для тіла підпрограми. Кнопками панелі **Programming** або клавішами «]» додати 3 рядки:

```
quad(a, b, c) := |
                 |
                 |
                 |
```

2. Ввести рядки тіла. Обчислення детермінанту та присвоєння його значення змінній D. Обчислення знаменника формули та присвоєння його значення змінній Z. Обчислення кореню.

$$\text{quad}(a, b, c) := \begin{cases} D \leftarrow b^2 - 4 \cdot a \cdot c \\ Z \leftarrow 2 \cdot a \\ \frac{-b + \sqrt{D}}{Z} \end{cases}$$


Програма *quad* має три параметри: коефіцієнти рівняння. Програма реалізує лінійний алгоритм, бо всі оператори виконуються завжди суворо послідовно.

### Алгоритми, що розгалужуються

У розгалужених алгоритмах присутні кілька гілок обчислювального процесу. Вибір конкретної гілки залежить від виконання (чи невиконання) заданих умов [9].

Наприклад, функція  $y(x)$  задається наступним виразом

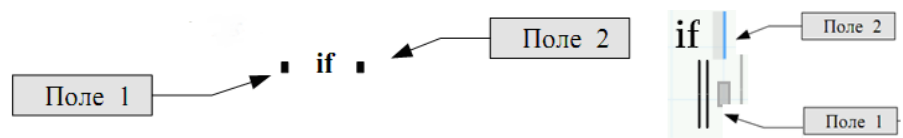
$$y(x) = \begin{cases} x^2, & x \leq 0; \\ \sqrt{x}, & x > 0. \end{cases} \quad (*)$$

Видно, що алгоритм обчислень містить дві гілки і вибір залежить від значення змінної  $x$ .

Для програмування алгоритмів, що розгалужуються підпрограми-функції в *MathCAD* використовують умовний оператор.

### Умовний оператор

Цей оператор використовується тільки в тілі програми-функції і для його введення необхідно клацнути на кнопці **if** панелі **Programming** або клавіші «}». На екрані з'являється конструкція з двома полями введення



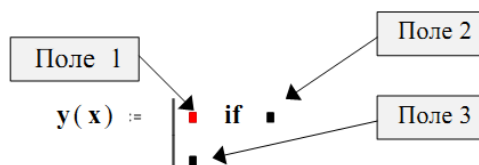
В поле 2 вводиться логічний вираз (у найпростішому випадку це вираз відношення). В поле 1 вводиться вираз (як правило, арифметичне), значення якого використовується, якщо логічний вираз приймає значення «1».

Умовний оператор реалізує «неповну» альтернативу. Тобто вказує, що потрібно виконати при виконанні умови. Для «повної» альтернативи застосовується оператор **otherwise**.

Вираз, що стоїть перед словом **otherwise**, виконується тільки в тому випадку, якщо не виконано задане перед цим умову.

В програмі можна використовувати кілька умовних операторів в сусідніх рядках з одним виразом перед словом **otherwise**.

Умовний оператор може знаходитися тільки всередині тіла програми-функції.



В поле 3 задається вираз, значення якого використовується, якщо логічний вираз дорівнює 0, наприклад – оператор **otherwise**. В поле 1 може бути розміщено декілька рядків. Для цього слід застосувати операцію вставки рядка **Add line**.

Наприклад, обчислення функції (\*) **y(x)** в підпрограмі можна виконати з використанням умовної функції або з умовним оператором:

$$y(x) := \left| \begin{array}{l} z \leftarrow \text{if}(x \leq 0, x^2, \sqrt{x}) \\ z \end{array} \right. \quad y(x) := \left| \begin{array}{l} x^2 \text{ if } x \leq 0 \\ \sqrt{x} \text{ otherwise} \end{array} \right.$$

```

y(x) := | if x <= 0
        | x^2
        | else
        | sqrt(x)
    
```

Наприклад, підпрограма-функція для обчислення функції **z(t)** за формулою

$$z(t) = \begin{cases} t^3, & t < -3; \\ \ln(t), & t > 4; \\ t^2, & -3 \leq t \leq 4. \end{cases}$$

може мати наступний вигляд:

$$z(t) := \begin{cases} t^3 & \text{if } t \leq -3 \\ t^2 & \text{if } -3 < t \leq 4 \\ \ln(t) & \text{otherwise} \end{cases}$$

```
z1(t) := || if t ≤ -3
          || t³
          || else if -3 < t ≤ 4
          || t²
          || else
          || ln(t)
```

Зауваження. Якщо в поле 3 ввести оператор **ln(t)** без слова **otherwise**, то цей оператор буде виконуватися завжди незалежно від виконання вище заданих умовних операторів. Для вірної відповіді слід ввести оператор

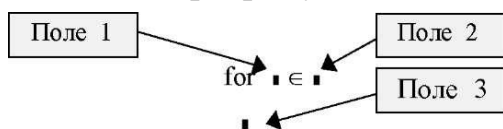
```
ln(t) if t > 4
```

### Циклічні алгоритми

Циклічні алгоритми (або цикли) містять повторювані обчислення, що залежать від деякої змінної. Така змінна називається параметром циклу, а самі повторювані обчислення складають тіло циклу.

#### Програмування зумовленого циклу

Для програмування таких циклів використовується оператор циклу **for**



Введення оператора проводиться клавішами *Ctrl + Shift + "*, кнопкою **for** панелі **Programming**.

В поле 1 вводиться ім'я змінної лічильника циклу, в поле 2 – діапазон значень параметра циклу. Діапазон значень вводиться аналогічно діапазону ранжованої змінної. В поле 3 вводяться оператори, що складають тіло циклу. Якщо одного рядка недостатньо, то додаткові поля введення (додаткові рядки) створюються процедурою **Add line**.

Наприклад. Функція *form\_tab1*, яка для **x**, що змінюється від -2 до 2 з кроком 0.5, розраховує значення  $f(x) = e^x \cdot \cos(2x)$  у вигляді вектора **y** ( $y_1 = f(-2)$ ,  $y_2 = f(-1.5)$ ) тощо, може мати наступний вигляд:

$$\text{form\_tab1}(x_{\text{start}}, x_{\text{stop}}, \text{step}) := \left| \begin{array}{l} i \leftarrow 1 \\ \text{for } x \in x_{\text{start}}, x_{\text{start}} + \text{step} .. x_{\text{stop}} \\ \quad \left| \begin{array}{l} y_i \leftarrow e^x \cdot \cos(2 \cdot x) \\ i \leftarrow i + 1 \end{array} \right. \\ y \end{array} \right.$$

У цьому варіанті опису програми-функції формальні параметри використовуються для визначення діапазону зміни параметра циклу (змінна  $x$ ). Для зміни індексу елемента масиву «у» вводиться змінна «і» цілого типу всередині циклу. Звернення до описаної підпрограми-функції може мати наступний вигляд

$$z := \text{form\_tab}(-2, 2, 0.5)$$

Кількість повторень визначається за формулою

$$N = \frac{x_k - x_0}{d} + 1 = \frac{2 - (-2)}{0.5} + 1 = 9$$

де  $x_k$ ,  $x_0$  – кінцеве і початкове значення параметра циклу,  $d$  – крок його зміни.

Опис програми-функції з циклом по кроках від 1 до  $N$  має наступний вигляд :

$$\text{form\_tab}(x_{\text{start}}, x_{\text{stop}}, \text{step}) := \left| \begin{array}{l} N \leftarrow \frac{x_{\text{stop}} - x_{\text{start}}}{\text{step}} + 1 \\ \text{for } i \in 0 .. N - 1 \\ \quad \left| \begin{array}{l} x \leftarrow x_{\text{start}} + i \cdot \text{step} \\ y_i \leftarrow e^x \cdot \cos(2 \cdot x) \end{array} \right. \end{array} \right.$$

Зауваження. Якщо значення індексів у елементів масиву змінюється починаючи з 1 (як в цьому прикладі), то початкове значення індексу необхідно встановити рівним 1 (системною змінною **ORIGIN** або пунктом меню **Tools-WorkSheet Option-Build in variables-Origin**).

## Програмування ітераційних циклів

Для програмування таких циклів використовується оператор циклу **while**.

Оператор циклу **while** виконується в такий спосіб: по-перше *MathCAD* перевіряє зазначену умову. Якщо вона істинна, то виконується тіло циклу і знову перевіряється умова. Якщо вона хибна, то цикл закінчується.

Для введення цього оператора необхідно клацнути на кнопці **while** панелі **Programming**. На екрані з'являються наступні елементи



Поле 1 призначено для розташування умови виконання циклу, поле 2 – для операторів тіла циклу.

В тілі циклу повинні бути присутніми оператори, які роблять умову циклу хибною, інакше цикл буде продовжуватися нескінченно.

Організація ітераційного циклу за допомогою оператора **while**, без додаткових засобів контролю може привести до зациклення, коли умова не буде виконана ніколи.

В *MathCAD* є декілька операторів, які призначені розв'язання таких ситуацій: вихід з циклу **break**, продовження **continue**, переривання **return**.

Необхідність застосування оператору **break** скоріш за все говорить про недолугий алгоритм розв'язання задачі.

Приклад. Підпрограма-функція, що реалізує ітераційну процедуру наближеного обчислення кореня квадратного з числа  $x$  з точністю  $\varepsilon$  за

ітераційним правилом  $x_{i+1} = \frac{x_i + \frac{x}{x_i}}{2}$ ,  $x \neq 0$ , з умовою закінчення розрахунку -  $|x_{i+1} - x_i| < \varepsilon$  може мати наступний вигляд:

```

sqrt(x, a) :=
  first ← x
  sec ← (first + x / first) / 2
  while |sec - first| > a
    first ← sec
    sec ← (first + x / first) / 2

```

Як видно з тексту програми-функції немає необхідності зберігати в пам'яті все при наближенні значення  $x_0, x_1, x_2, \dots$  тощо. Досить зберігати два: попереднє («старе») значення **first** і наступне **sec**.

Звернення до описаної програми буде мати вигляд

**sqrt(9, 0.0001)=3 sqrt(25, 0.0001)=5** .

### Вихід з циклу

Оператор **break** дозволяє вийти з циклу. Застосовується разом з умовним оператором. При виконанні заданої умови цикл переривається, керування передається наступному за циклом рядку.

Введення оператора **break** проводиться клавішами *Ctrl+{* або кнопкою **break** панелі **Programming** (не можна вводити оператор **break** з клавіатури по символах). Оператор **break** розташовується в лівому полі введення умовного оператор **if**, а в правому розміщується умова, за виконання якої відбувається припинення роботи циклу або програми.

Наступний приклад показує написання програми прикладу з оператором **break**.

```

sqrt(x, a) :=
  first ← x
  sec ← (first + x / first) / 2
  while 1
    first ← sec
    sec ← (first + x / first) / 2
    break if |first - sec| < a

```



У цій програмі застосовано нескінчений цикл. Для виходу з циклу за умови  $|x_{i+1} - x_i| < \varepsilon$  або в термінах програми  $|\text{sec} - \text{first}| < a$ , використано оператор **break**.

### Оператор продовження

Зазвичай оператор **continue** використовується для продовження виконання циклу шляхом повернення на початок тіла циклу без виконання рядків, які розташовані після оператора **continue**. Наступний приклад пояснює роботу цього оператора.

Наприклад, потрібно сформувати з вихідного вектору  $x$  результуючий вектор  $y$ . Вектор  $y$  повинен містити тільки додатні елементи вектору  $x$ .

### РОЗВ'ЯЗАННЯ

Підпрограма може мати наступний вигляд:

```

positiv(x) :=
  ix ← -1
  iy ← 0
  while ix < last(x)
    ix ← ix + 1
    continue if xix ≤ 0
    yiy ← xix
    iy ← iy + 1
  y

```

В тілі програми-функції використовується вбудована функція **last(x)**, яка визначає індекс останнього елемента вихідного масиву  $x$ . Якщо черговий елемент  $x_i$  від'ємний, то пропускаються всі оператори тіла циклу нижче умови (оператори, що обраховують елемент результуючого вектора  $y$ ) і тіло циклу повторюється при новому значенні параметра циклу  $i$ .

### Оператор переривання

Оператор **return <Вираз 1>** перериває виконання програми-функції і повертає значення операндам **<Вираз 1>**, що стоїть за ним. Наприклад, програма для знаходження першого додатного елемента масиву  $x$  може мати наступний вигляд:

```

positiv(x) := for i ∈ 1..last(x)
  return xi if xi > 0

```

## Оператор помилки

Оператор **аж error** є засобом обробки помилок, які виникають при виконанні тих чи інших обчислень, і записується у вигляді:

**<Вираз 1> on error <Вираз 2> .**

Оператор виконує <вираз 2>. Якщо при виконанні <виразу 2> виникає помилка, то виконується <вираз 1>.

Наприклад, обчислення функції  $\frac{\sin(x)}{x}$  може мати наступний вигляд

$s(x) := 1 \text{ on error } \frac{\sin(x)}{x}$

### 3.2.2. Завдання для самостійного виконання

*Завдання 3.2.1.* Розробити підпрограму для знаходження всіх коренів квадратного рівняння та використати її як зовнішнє посилання.

*Завдання 3.2.2.* Розробити підпрограму з трьома аргументами a,b,c. Функція замінює числа їх квадратами якщо  $a \leq b \leq c$ , замінює числа на максимальне значення з чисел, якщо  $a > b > c$ , змінює знаки чисел в інших випадках. Використати її як зовнішнє посилання

*Завдання 3.2.3.* Розробити підпрограму з двома аргументами – координатами точки x, y. Функція обраховує номер квадранта, в якому знаходиться точка.

*Завдання 3.2.4.* Розробити підпрограму з трьома аргументами – довжинами сторін трикутника a, b, c. Функція повертає 3 якщо трикутник рівносторонній, 2 якщо – рівнобедрений, 1 – всі сторони мають різну довжину.

*Завдання 3.2.5.* Розробити підпрограму для обрахунку нескінченної суми послідовності  $a_i = \frac{1}{\sqrt{i}}$ . Сумування закінчити за умови  $|a_i| < \varepsilon$ .

*Завдання 3.2.6.* В матриці визначити індекси значення елементів, які найближчі до заданого числа.

*Завдання 3.2.7.* Сформувати вектор, і-й елемент якого дорівнює кількості додатних елементів в і-му рядку вихідної матриці.

*Завдання 3.2.8.* В векторі  $x$  порахувати добуток додатних елементів, суму елементів в діапазоні  $a < x_i < b$ .

*Завдання 3.2.9.* Аргумент функції  $f(x)$  визначено вектором  $x$ . Визначити максимальне значення функції, номер елемента вектора  $x$ , який відповідає максимальному значенню функції.

*Завдання 3.2.10.* В матриці  $A$  поміняти місцями максимальний та мінімальний елементи.

*Завдання 3.2.11.* В матриці на місце першого елемента  $i$ -го рядка записати суму елементів цього рядка.

*Завдання 3.2.12.* За даними з матриці  $A$  сформувати вектор  $x$ .  $i$ -й елемент вектору  $x$  має дорівнювати номеру стовбця матриці  $A$ , в якому на  $i$ -му рядку матриці знаходиться мінімальний елемент.

*Завдання 3.2.13.* В матриці переставити стовпці в порядку збільшення суми елементів стовпців.

*Завдання 3.2.14.* В матриці визначити максимальний індекс стовпців та мінімальний рядковий індекс від'ємних елементів.

### **Практикум 3.3. Двомірні графіки**

#### **3.3.1. Теоретичні положення**

На практиці виникають задачі, які потребують візуалізації даних в графічному вигляді. Наприклад:

- огляд характерних точок функції для експрес-аналізу результатів,
- дослідження функції для виявлення характерних точок, точок перетину осі абсцис тощо,
- аналіз вихідних табличних даних, знаходження по табличним даним наближаючої аналітичної залежності, оцінка ступеню наближення.

В систему вбудовані засоби для виведення графічної інформації трьох видів: двомірних графіків, тримірних графіків, малюнків. Для поліпшення візуального сприйняття можливо зі статичних графіків зробити анімацію.

Для графічної інформації призначено спеціальну графічну область.

Керування графічною областю здійснюється за допомогою «гарячих клавіш», через пункт меню **Insert-Graph** або через кнопку панель графіки. В *MathCAD Prime* створення графічної області проводиться «гарячими клавішами» та стрічкою **Plots-Insert Plot** (рис. 3.30).

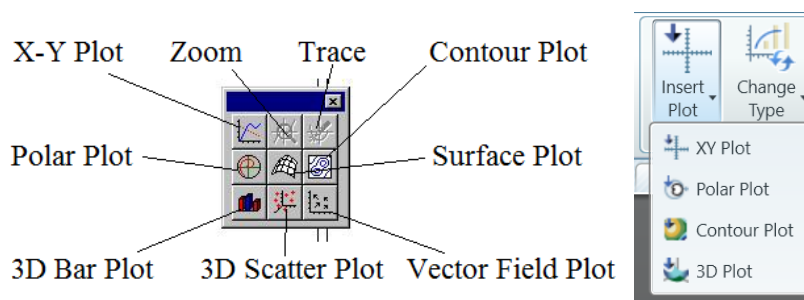


Рис. 3.30. Панель графіки

Плоскі графіки призначені для зображення функцій, виразів одного аргументу, векторів. Система будує графіки функцій однієї змінної в декартових  $y=f(x)$  та полярних  $\rho=f(\varphi)$  координатах, де  $y = \rho \sin(\varphi)$ ,  $x = \rho \cos(\varphi)$ .

Залежності можуть бути записані в абсолютній або в параметричній формі  $y = f(t)$ ,  $x = \varphi(t)$  як залежності від однієї спільної змінної.

### Графіки в декартових координатах

Активація графічної області плоских декартових графіків виконується натисканням клавіш *Shift+@*. При цьому на екрані відображується умовне поле графіка у вигляді рамки й прямокутників біля осей графіка для введення значень (рис. 3.31).

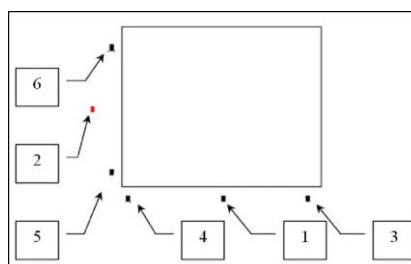


Рис. 3.31. Маркери області двомірної декартової графіки

Середні маркери призначені для введення ранжованої змінної-аргументу по осі абсцис (поле 1) та функції/й-користувача, виразів, які необхідно відобразити, по осі ординат (поле 2). Для побудови графіків декількох функцій їхні імена вводяться в поле 2 через кому.

Крайні маркери призначені для визначення границь графіків: маркери 3,4 визначають межі аргументу, маркери 5,6 – функції. Якщо межі графіка не задані, пакет проставляє їх автоматично.

За замовчанням графік будується в діапазоні аргументу від -10 до 10 із кроком 1.

Обов'язковим є заповнення тільки поля 2. Введення тільки імені функції в поле 2 забезпечує «швидку» побудову графіка з параметрами за замовчанням.

Наприклад, для того щоб відобразити графік функції  $y=e^{-x}\cdot\cos(5x)$  у діапазоні  $x \in [0, 5]$  із кроком 0.1, потрібно:

1. Визначити функцію користувача  $y(x):=\exp(-x)\cdot\cos(5\cdot x)$ .

2. У потрібному місці вставити графічну область декартового графіка знаком “@” або іншим способом. У поле 2 ввести ім'я функції  $y(x)$ . Графік набуде вигляд (рис. 3.32). Діапазон аргументу -10...10 та відповідні значення в полях 5,6: -20 та 20,- система визначила автоматично.

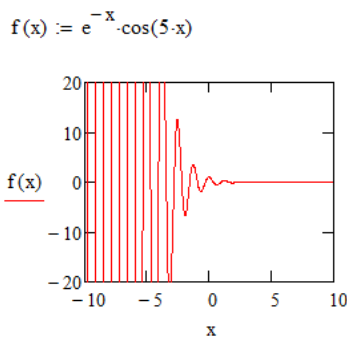
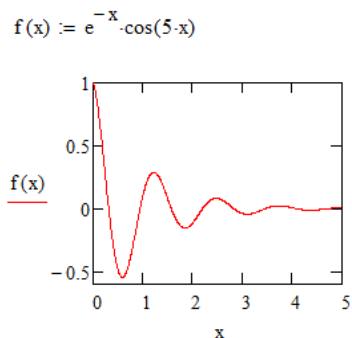


Рис. 3.32. Графік за замовчанням

3. Для зміни меж аргументу слід змінити значення маркерів 3,4 на 0 та 0.5 відповідно. Для зміни меж значень графіка по осі ординат – маркера 5 на -0.5. Графік набуде вигляд (рис. 3.33).



### Рис. 3.33. Відкоригований графік

Примітка. Для усунення необхідності додаткових дій доцільно визначати діапазон змін аргументу через ранжовану змінну.

В області двомірних графіків можна візуалізувати табличні (векторні) дані та будувати *параметричні графіки*. При роботі з векторами аргументом графіка є змінна індексу вектора або один із векторів. Другий вектор є ординатою графіка.

На *параметричному графіку* по осях координат задаються дві функції, які залежать від одного аргументу. Використання замість функцій векторів дозволяє побудувати зображення точкової діаграми. Один вектор повинен містити координати  $x$  точок, другий – координати  $y$ .

Параметричні графіки зручно використовувати для побудови графіків функцій, які мають декілька значень функції при одному аргументі. Зазвичай такі функції описують замкнені геометричні фігури на кшталт кола.

Наприклад, система зображує в декартових координатах тільки одну гілку графіка (рис. 3.34а) функції  $y = \sin\left(3\frac{\arcsin(x)}{2}\right)$ . Якщо записати цю функцію в параметричному вигляді  $a := \sin(2t)$   $b := \sin(3t)$ , то графік стає значно інформативнішим (рис. 3.34б).

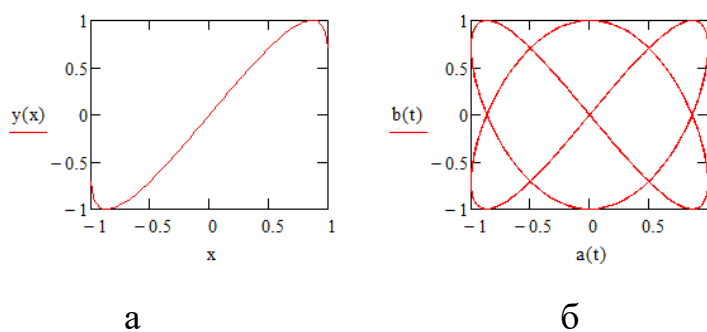


Рис. 3.34. Графік: а – абсолютної функції, б – параметричної функції

Модифікацію плоского графіка можна провести за допомогою локального форматування двома способами: двократним клацанням кнопки миші при розташуванні курсору в області графіка або через меню **Format - Graph**. В обох випадках активізується меню плоских графіків (рис. 3.35).

Вікно меню графіків має наступні вкладники:

*X-Y Axes* – визначає вигляд осей графіка;

*Traces* – визначає вигляд ліній самого графіка;

*Labels* – визначає вигляд пояснюючих написів по осях графіка;

*Defaults* – визначає значення за замовчанням;

*Number Format* – визначає формат виведення чисельних значень на графіку.

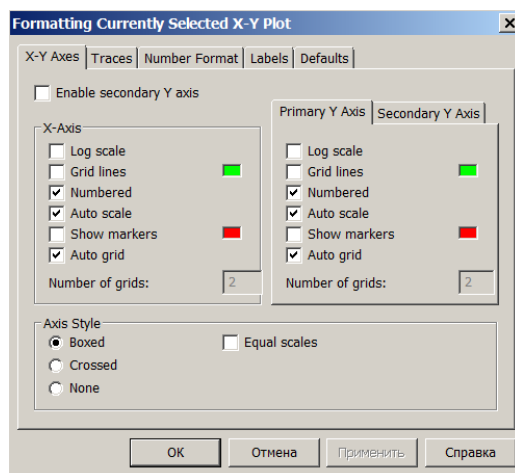


Рис. 3.35. Вікно форматування двовірних графіків

Поля закладки **X-Y Axes**:

*Enable second Y axis* додає другу вісь «Y» для відображення двох графіків з різними значеннями ординат. При малюванні кількох графіків перший малюється відносно першої осі, всі інші – відносно другої.

*Log Scale* задає логарифмічний масштаб осей. Масштаб задається по кожній осі незалежно.

*Grid Lines* керує видимістю допоміжних ліній сітки на полі графіка.

*Show markers* додає одну або дві лінії сітки з визначеним положенням.

*Numbered* керує виведенням проміжних чисел усередині діапазону по лініях сітки.

*Autoscale* задає режим автоматичного масштабування графіка.

*Autogrid* задає автоматичне нанесення сітки.

*Axis Style* керує виглядом осей: *Boxed* – прямокутник навколо графіка; *Crossed* – осі у вигляді хреста; *None* – відсутність осей; *Equal scale* – осі рівного масштабу.

Поля вкладники **Labels**: *Title* – задає титульний напис до графіка; *X-Axis* – задає напис осі X; *Y-Axis* – задає напис осі Y.

Поля вкладники **Traces**:

*Legend* керує виведенням опису ліній графіка.

*Symbol* задає символ, що буде зображено на лінії графіка: хрест, коло, квадрат.

*Symbol Frequency* задає частоту повторення символів. Значення показує кількість точок графіка між символами.

*Symbol Weight* задає товщину символів

*Line* задає візерунок лінії: суцільна, точкова, штрихова тощо.

*Color* задає колір лінії.

*Type* задає тип лінії графіка: звичайна лінія (*line*), стовпова діаграма (*bar*), лінійна діаграма (*stem*), лінія з точок (*points*), лінія сходів (*step*). Для двох ліній можна застосувати тип «похибка». При цьому на графіку буде заштриховано область, що окреслена лініями графіка. Тип «draw» дозволяє малювати графік без продовження до осей графіка.

*Weight* задає товщину лінії.

## Графіки в полярних координатах

Активація графічної області плоских полярних графіків виконується натисканням клавіш *Ctrl+7*. При цьому на екрані відображується умовне



поле графіка у вигляді рамки з маркерами біля контурного кола для введення значень (рис. 3.36). Нижній маркер 1 призначено для введення імен ранжованої змінної-аргументу, лівий 2 – для введення функцій/виразів, графіки яких необхідно відобразити. За замовчанням приймається, що діапазон зміни аргументу становить  $0 - 2\pi$ . Маркери 3, 4 призначені для обмежень значень ординат по радіусу: 3 – максимальне значення радіуса, 4 – мінімальне.

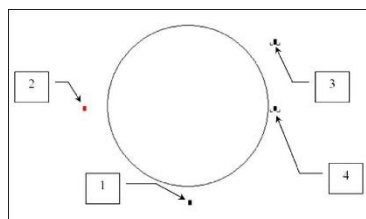


Рис. 3.36. Шаблон графіка в полярних координатах

Модифікацію графіка в полярних координатах можна провести за допомогою локального форматування аналогічно форматуванню графіка в декартових координатах. Параметри графіка редагуються у діалоговому вікні полярних графіків.

Вікно має наступні вкладники: **Polar Axes**, **Traces**, **Labels**, **Defaults**, **Number Format**. Зміст дій та поля вкладинок в цілому співпадають з полями вікна декартових графіків.

Поле **Show markers** за документацією виробника повинно виводити додаткові лінії сітки по визначених двох додаткових радіусах. На практиці не працює.

Поле **Show negative radii** дозволяє використовувати вирази з від'ємним значенням радіуса.

Наприклад, для того щоб відобразити графік функції

$$r(\varphi) = 1 + \frac{\sin(3\varphi + \pi)}{2} \quad \text{в полярних координатах, слід:}$$

- Визначити функцію користувача  $r(\varphi)$ ;
- У потрібному місці вставити графічну область полярного графіка клавішами **Ctrl+7** або іншим способом. У поле 2 ввести ім'я функції

$r(t)$ . Графік набуде вигляд, зображений на рис. 3.37. Діапазон аргументу  $0 \dots 2\pi$  та відповідні значення в полі 1 система визначила автоматично.

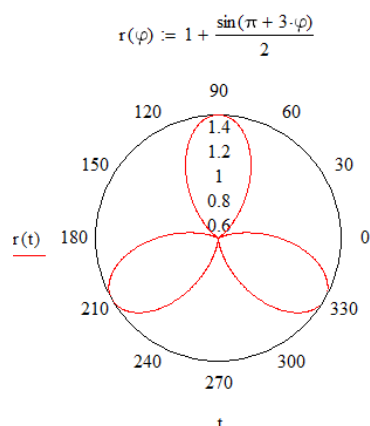


Рис. 3.37. Графік в полярних координатах

Для спрощення розрахунків можна застосовувати функції переведення декартових координат в полярні та навпаки: **xy2pol (<arg>)** та **pol2xy (<arg>)**. Аргументом можуть бути два числа – координати  $x, y$ /радіус, кут або двоелементний вектор. Перший елемент вектора містить значення  $x$ /радіуса, другий –  $y$ /кута в радіанах. Результатом є вектор полярних координат/декартових координат. Перший елемент вектора містить значення радіуса/ $x$ , другий – кута в радіанах/ $y$ .

Наприклад, звернення до функції з аргументами «1, 1» або  $\begin{bmatrix} 1 \\ 1 \end{bmatrix}$  виведе

результат  $xy2pol(1, 1) = \begin{pmatrix} 1.414 \\ 0.785 \end{pmatrix}$ .

### Додаткові засоби роботи з плоскими графіками

#### Логарифмічний масштаб осей

Логарифмічний масштаб слушно використовувати:

- коли необхідно мати детальну інформацію про всі ділянки графіка, діапазон змін аргументу або значень функції дуже великий;
- коли на одному полі необхідно відобразити графіки, які мають дуже різні значення.

Логарифмічний формат задається для тої осі, по якій очікується великий розбіг значень. В логарифмічному масштабі розтягується область маленьких значень та звужується для великих.

Приклад застосування логарифмічних масштабів наведено в п. 3.6.

Для спрощення форматування даних в логарифмічному масштабі призначені функції **logspace**, **logpts**.

Функція **logspace(min,max,npts)** повертає вектор, **npts** елементів якого розподілені за логарифмічним законом від **min** до **max**. Наприклад, виклик **logspace(0.1,1,9)** генерує вектор [0.1, 0.132, 0.173, 0.228, 0.3, 0.395, 0.52, 0.684, 0.9].

Функція **logpts(minexp,dec,dnpts)** повертає вектор, **dec · dnpts** елементів якого лінійно розподілені від  $10^{\text{minexp}}$  на **dec** декад з **dnpts** інтервалами на декаду. Функція не розраховує останню точку декади. Наприклад, **logpts(-1,1,9)** генерує вектор [0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9]. Початок  $10^{-1}=0.1$ , одна декада від 0.1 до 1 з дев'ятьма точками. Виклик **logpts(-1,1,10)** генерує не [0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0], а [0.1, 0.19, 0.28, 0.37, 0.46, 0.55, 0.64, 0.73, 0.82, 0.91].

Приклад дії функцій наведено на рис. 3.38.

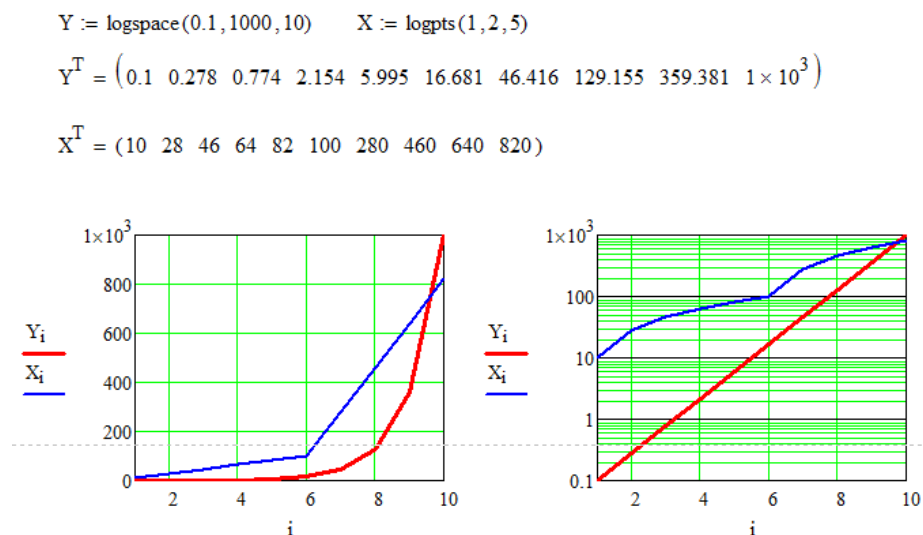



Рис. 3.38. Дія функцій **logspace**, **logpts**

Детальний кількісний аналіз графіків

Для дослідження поведінки графіків та отримання поточних кількісних даних зручно використовувати режими «збільшення» (*Zoom*) та «аналіз» (*Trace*). Доступ до цих режимів реалізовано через кнопки панелі **Graph** та поля **Zoom**, **Trace** контекстного меню графічної області. Керування режимами здійснюється через відповідні діалогові вікна.

Для визначення області збільшення можна задати її координати в полях «x», «y» у вікні збільшення або просто відмітити прямокутну область на графіку мишею з натиснутою лівою кнопкою. При роботі мишею координати синхронно відображаються в відповідних полях вікна збільшення. Натискання кнопки **ОК** призводить до перемалювання графіка в новому масштабі. Завжди можна повернутися до первісного вигляду графіка натисканням кнопки  у вікні збільшення.

У вікні трасування синхронно з рухами миші по полю графіка відображаються її координати. Для того, щоб миша рухалася тільки по лінії графіка слід увімкнути перемикач **Track data points** у вікні трасування.

Активація графічної області плоских декартових графіків в *MathCAD Prime* виконується з панелі **Plots-Traces-Insert Plots-XY Plot** стрічки або натисканням клавіш *Ctrl+2*. При цьому на екрані відображується умовне поле графіка у вигляді рамки з маркерами біля осей графіка для введення значень (рис. 3.39).

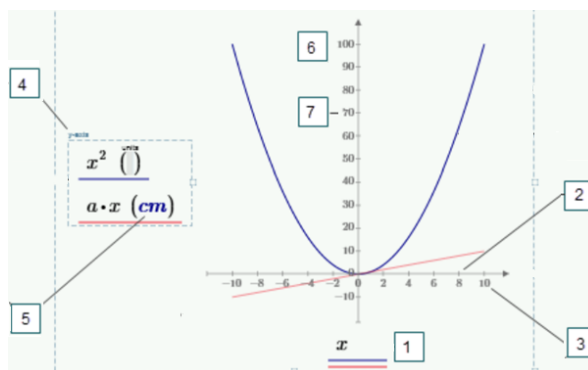


Рис. 3.39. Поле графіка в декартових координатах з маркерами: 1 – **X axis** аргументів, 2 – кроку міток осі X, 3 – максимального значення

по осі X, 4 – **Y axis** функції, 5 – одиниць вимірювання, 6 – максимального значення по осі Y, 7 – кроку міток по осі Y

Маркер **Y axis** 4 призначений для введення імені функції користувача, графік якої планується накреслити. Маркер 1 **X axis** призначений для введення імені ранжованої змінної-аргументу або вектора-аргументу графіка. Для побудови графіків декількох функцій слід



використати кнопку *Plots-Traces-Add Trace* стрічки.

За замовчанням двомірний графік в декартових координатах будується в діапазоні аргументу від -10 до 10 з кількістю точок 500. Імені аргументу за замовчанням немає.

Редагування границь графіків проводиться безпосередньою зміною крайніх значень по осях.

Редагування плоских декартових графіків проводиться безпосередньо на полі графіка та засобами закладки **Plots** стрічки (рис. 3.40).

Маркери 1, 4 можуть бути пересунені в потрібне користувачеві місце.

Розташування осей можна змінити простим перетягуванням їх по полю графіка.

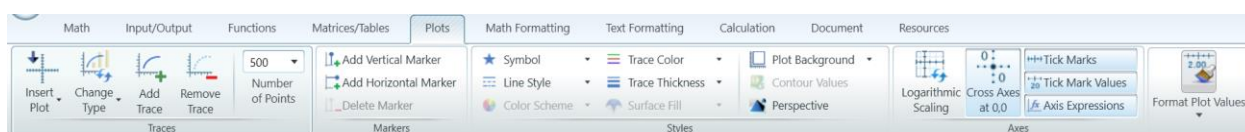


Рис. 3.40. Стрічка керування графіками

Активація графічної області полярних графіків в *MathCAD Prime* виконується натисканням клавіш *Ctrl+4* або стрічкою **Plots-Insert Plot- Polar Plot**. На екран виводиться шаблон полярного графіка з маркерами для введення значень (рис. 3.41). Зміст і поведінка маркерів є такими ж, як і на полі графіків в декартових координатах. Маркер **Radial Axes** 1 слугує для введення функції/виразу, графіки якої необхідно відобразити. Маркер **Angular Axes** 3 призначено для введення імені ранжованої змінної-аргументу. За замовчанням діапазон зміни аргументу

становить  $0 - 2\pi$ . При наведенні миші на маркери 1, 3 відкриваються маркери для введення одиниць вимірювання.

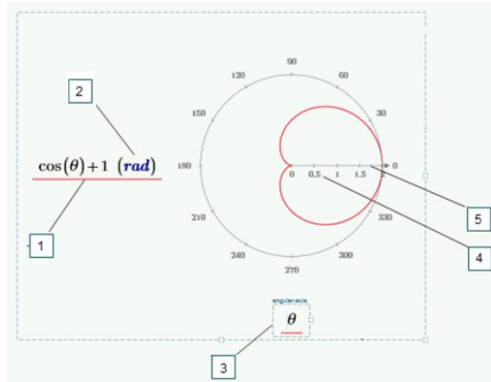
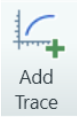


Рис. 3.41. Шаблон графіка в полярних координатах

Для додавання графіків на поле використовується кнопка  **Plots-  
Traces-Add Trace** стрічки.

Редагування полярних графіків проводиться безпосередньо на полі графіка (рис. 3.41) та засобами закладки **Plots** стрічки.

### 3.3.2. Приклади та завдання для самостійного виконання

*Приклад 3.3.1.* Побудувати графіки функції  $Y(t) := \sin(t)$ ,  $Y1(t) := t^2$ . За допомогою функцій збільшення та трасування знайти точки перетину графіків.

#### РОЗВ'ЯЗАННЯ

1. Визначити функції користувача  $y(t) := \sin(t)$  та  $y1(t) := t^2$ .
2. Визначити діапазон аргументу ранжованою змінною  $x := -0.5, -0.49.. 1$ .
3. Додати графічну область. В маркері аргументу вставити ім'я ранжованої змінної, в маркері функцій – через кому імена введених функцій користувача (рис. 3.41).

З графіка видно, що криві мають дві точки перетину. Перша точка очевидна –  $(0,0)$ .

Для уточнення значення другої застосуємо режим збільшення.

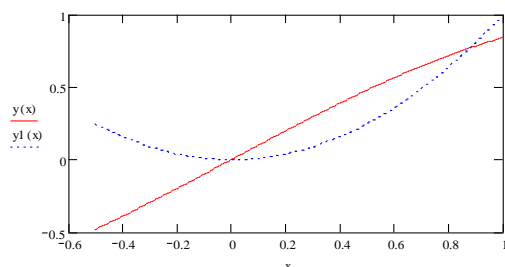


Рис. 3.42. Графіки функцій прикладу 3.3.1

4. Обрати поле **Zoom** контекстного меню графічної області. Вибрати область графіків навколо точки перетину. Можливе зображення наведено на рис. 3.43.

Для уточнення значення координат точки перетину застосуємо режим слідування (*trace*).

5. Обрати поле **Trace** контекстного меню графічної області. Навести курсор на точку перетину. В полях «X-Value», «Y-Value» спостерігати координати точки перетину. Можливе зображення наведено на рис. 3.44.

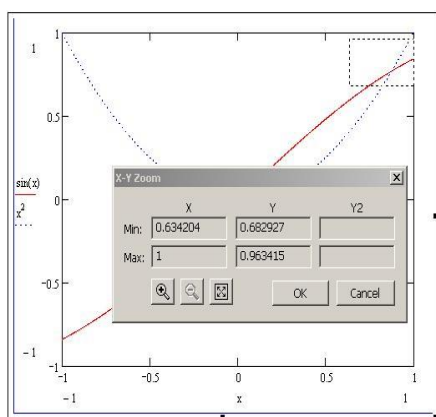


Рис. 3.43. Режим **Zoom**

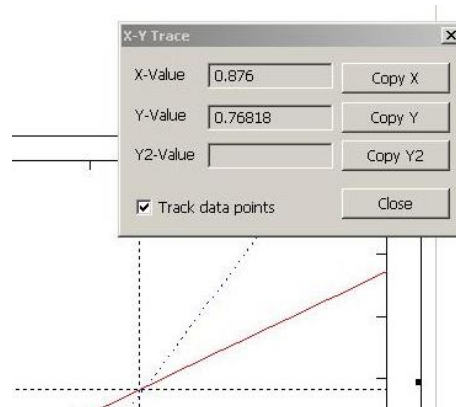


Рис. 3.44. Режим **Trace**

*Приклад 3.3.2.* Побудувати в одній графічній області графіки спектральних щільностей генераційно–рекомбінаційного  $U_{GR}(f) = 10^{-12} \frac{RU}{f}$  та струмового  $U_s = 2qR^2$ , де  $q = 1.6 \cdot 10^{-19}$  кл – заряд електрону, шумів перетворювача з напругою  $U = 10$ В, опором  $R = 50$ кОм у смузі частот 0.1 Гц ... 100 Гц.

## РОЗВ'ЯЗАННЯ

$U := 10$  - значення напруги  
 $R := 5 \cdot 10^4$  - значення опору перетворювача  
 $R_n := R$  - значення опору навантаження

$Uf(x) := 10^{-12} \cdot \frac{R_n^2 \cdot U}{x \cdot R}$  - спектральна щільність рекомбінаційного шуму

$Ut(x) := 2 \cdot 1.6 \cdot 10^{-19} \cdot R^2$  - спектральна щільність струмового шуму

$f := 0.1, 0.5 \dots 100$

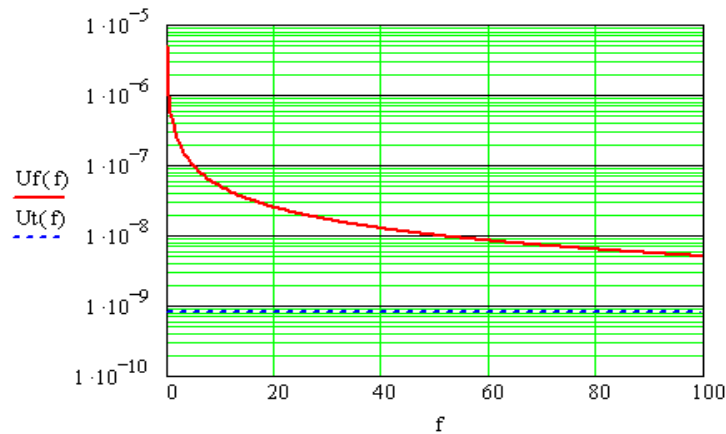


Рис. 3.45. Графік прикладу 3.3.2

*Завдання 3.3.1.* Для функції в одній графічній області побудувати графіки самої функції, інтегралу від 0.5 до 2, першої та другої похідної

а)  $\sin(x) \cos(x)$  б)  $\exp\left(-\frac{x}{2}\right) + 2\sin\left(\frac{x}{2}\right)$ .

*Завдання 3.3.2.* Побудувати графіки в двох вікнах, в одному вікні, в одному вікні на двох осях функцій в діапазоні аргументу  $0..2\pi$

$3 - \cos(x^2)$      $2.5 \sin(2t)$  .

*Завдання 3.3.3.* Побудувати на одному полі графіки за формулою Планка (приклад 3.5.4) для  $T=1000, 1500, 2000$  в діапазоні  $[0.1 \dots 30]$ . Використати одну функцію користувача.

*Завдання 3.3.4.* Побудувати на одному полі в діапазоні аргументу  $[0 \dots 6]$  графіки функцій  $y=x$  та  $y=x^2$  та позначити область, яка лежить між графіками функцій та її межі.



**Завдання 3.3.5.** Побудувати графік функцій  $r=2(1-\cos(\varphi))$ ,  $r=3\sqrt{2\cos(2\varphi)}$  в полярних координатах.

**Завдання 3.3.6.** Намалювати коло з радіусом 10 та центром  $p(1,3)$ .

**Завдання 3.3.7.** Результати розрахунку координат точок зберігаються в матриці Coords. Перший стовпець уміщує координати X точок, другий – Y. Побудувати точкову діаграму (розподіл точок на площині).

	1	-8
	2	6
	3	5
	-4	4.2
	5	2
Coords :=	6	0
	-7	9.99
	8	-6
	9	2
	10	1

**Завдання 3.3.8.** За допомогою трасування визначити точки перетину функцій:  $y=2\cos(x)$ ,  $y=2\sin(x)^2$  та  $y=x$  для  $x [0..\pi]$ .

## Практикум 3.4. Тримірні графіки

### 3.4.1. Теоретичні положення

Активация графічної області 3D графіків здійснюється через пункт меню **Insert-Graph** або натисненням клавіш **CTRL+2**.

У місці маркера графіка для побудови 3D графіка необхідно ввести ім'я матриці/матриць, для якої передбачається відображення у вузлах, або ім'я функції/функцій двох змінних.

Зображення можна повернути та змінити його масштаб. Зсув миші з натиснутою лівою кнопкою синхронно повертає зображення. Зсув миші з натиснутими лівою кнопкою та клавішею **Ctrl** або поворот колеса миші наближає або віддаляє зображення.

### Керування виглядом просторових графіків

Виклик вікна зміни параметрів 3D графіка (рис. 3.46) проводиться подвійним щигликом лівої кнопки миші на полі графіка або з меню **Format - Graph**.

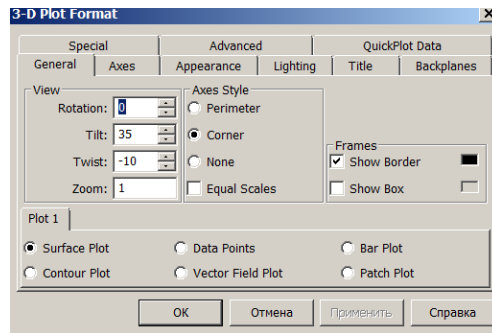


Рис. 3.46. Вікно форматування просторових графіків

Вікно має наступні вкладинки:

**General** – визначення точки зору, виду графіка;

**Axes** – визначення вигляду координатних осей;

**Appearance** – визначення кольорів, вигляду точок графіка;

**Lighting** – визначення додаткового освітлення;

**Title** – визначення титульних написів;

**Backplanes** – визначення сітки, кольору фонових площин;

**Special** – визначення типу ліній, контурних ліній, стовпових діаграм;

**Advanced** – установки прозорості, «задимлення», якості друку;

**QuickPlotData** – зміна діапазонів аргументів, системи координат.

Поля вікна тримірної графіки

**General** Розділ **View** задає орієнтацію та тип зображення.

*Rotation* – поворот. Задає кут повороту осей  $XU$  горизонтальної площини  $XOY$  відносно вертикальної осі  $Z$  у діапазоні  $-360^0 \dots 360^0$ . Нуль градусів відповідає  $YOZ$  площині, дев'яносто градусів –  $XOZ$ .

*Tilt* – нахил. Задає кут підвищення спостерігача відносно горизонтальної площини  $XOY$  у діапазоні  $0^0 \dots 180^0$ . Нуль градусів відповідає горизонтальній проекції площини (вид збоку), дев'яносто градусів - вертикальній проекції (вид згори).

*Twist* – кручення. Задає кут повороту осі Z у діапазоні  $360^0 \dots 360^0$  після повороту та нахилу.

*Zoom* – масштаб у діапазоні 0.0001 ... 10000.

Розділ **Axes Style** визначає стиль осей та рамки області:

*Perimeter* – по периметру; *Corner* – осі в куті, *None* – без осей, *Equal Scales* – рівні масштаби по всім осям.

Розділ **Frames** визначає рамку навколо графіка:

*Show Border* – прямокутна рамка навколо осей, *Show Box* – паралелепіпед навколо графіка.

Розділ **Plot** визначає тип графіка: *Surface Plot* – графік поверхні, *Contour Plot* – графік ізоліній (контурний вид знизу), *Data Points* – точковий графік, *VectorField Plot* – графік векторного поля, *Bar Plot* – діаграма, *Patch Plot* – діаграма рівнів.

## Appearance

*Fill options* – керує фарбуванням графіка:

*Fill Surface* – заповнення кольором за схемою з вкладники *Color Options*, *Fill Contours* – заповнення кольором у напрямку з вкладники *Special*, *No Fill* – без фарбування, *Alternate Mesh* – заповнення кольоровими трикутниками, *Smooth Shading* – покращене фарбування.

*Lines options* – керує видом відображення ліній поверхні:

*Weight* – товщина ліній, *Wireframe* – у вигляді каркаса із дроту, *Contour* – з контурними лініями визначеної площини, *No Lines* – у вигляді кольорового твердого тіла, коли сховані лінії не видно, *Hide Lines* – у вигляді твердого тіла (*No fill*), коли сховані лінії не видно.

*Points Options* – визначає вигляд позначок на лініях графіка:

*Draw Points* – вмикає видимість позначок, *Symbol* – визначає тип позначки (точка, хрест, ромб), *Size* – визначає розмір позначки.

*Color options* визначає тип фарбування: *Colormap* – градієнтне

фарбування з параметрами з вкладинки *Advanced*: райдуга, відтінки сірого, синього, зеленого, червоного тощо, *Solid Color* – суцільний колір.

**Axis** Вкладинка **Axis** визначає вигляд осей X, Y, Z. Зміст є аналогічним вкладинці двомірних графіків: сітка, маркери осей, колір.

**Special** Розділ **Line Style** надає випадаючий список типів ліній сітки та контуру: суцільна (*solid*), пунктирна (*dotted*), штрихова (*dashed*), штрих пунктирна (*da-dot*).

Розділ **Contour** керує виглядом контурного графіка: фарбуванням *Fill*, виведенням ізоліній *Draw Lines* автоматично *Auto Contour* чи у визначеній кількості з кількісними значеннями *Number*.

**Advanced** Розділ **Advanced View Options** вмикає режими туману *Enable Fog*, та перспективної проєкції *Perspective* на визначеній відстані *Viewing Distance*. Поле *Vertical Scale* у відносних координатах визначає вертикальний масштаб зображення.

Розділ **Plot** визначає оптичні якості поверхні: відбиття *Shininess*, прозорість *Transparency*, відступ зафарбованої ділянки від лінії сітки *Polygon Offset*.

Розділ **Colormap** керує напрямом градієнтного фарбування та його типом.

Розділ **Printing** керує якістю виведення зображень на принтер.

## Графіки поверхні

В основі відображення 3D графіків лежать матричні перетворення. Система відображає саме елементи матриці. При введенні в графічне поле імені функції двох змінних система автоматично застосовує вбудовану функцію **CreateMesh**, яка створює матрицю з функції з аргументами за

замовчанням. Визначення аргументів ранжованою змінною не впливає на діапазон зображення. *Коригування можливо тільки в ручному режимі в діалоговому вікні форматування 3D графіка* (рис. 3.46).

Примітка. Введення замість імені розрахункової залежності не дозволяється.

**CreateMesh(function, [s0, s1, t0, t1], [sgrid, tgrid], [fmap])**

Функція повертає три матриці, які в параметричному вигляді відображають координати  $x$ ,  $y$ ,  $z$  функції **function** в інтервалах **[s0-s1]** для першої координати, **[t0-t1]** – для другої, з **sgrid**, **tgrid** значеннями по кожній координаті.

Первинна функція **function** може бути записана в звичайному вигляді  $f(x, y) = \sin(x) + \cos(y)$  або параметричному вигляді як вектор-стовпець з трьох елементів, наприклад,

$$f(x, y) = \begin{bmatrix} x \\ y \\ \sin(x) + \cos(y) \end{bmatrix}.$$

Параметр **fmap** задається коли вихідна функція визначена не в декартових координатах. Він являє собою триелементний вектор. Елементами вектора є функції перерахування координат в декартові або ім'я функції перерахунку координат.

Примітка. Для перерахунку координат зручно використовувати вбудовані функції.

Усі параметри, окрім імені функції, не є обов'язковими. За замовчанням по обох аргументах діапазон:  $-5 \dots 5$ , число точок - 20. Зміною початку та кінця діапазону можна створювати графіки поверхонь із вирізом.

Наприклад, для швидкої побудови графіка функції двох змінних слід:

1. Визначити функцію користувача  $f(x, y) = x^2 + y^2$ ;
2. Активувати графічне поле клавішами *Ctrl+2* або в інший спосіб. На місце маркера вставити ім'я функції  $f$ .

Аналогічного результату можна досягти за допомогою функції **CreateMesh**. Порівняльні зображення наведені на рис. 3.47.

Наприклад, виклик функції для побудови графіка гіперболічного параболоїда  $F(x, y) = x^2 - y^2$  для  $-2 < x < 2$  та  $-3 < y < 3$  та кількістю точок 50 по кожній змінній має наступний вигляд

**F(x, y) := x<sup>2</sup> - y<sup>2</sup>    B := CreateMesh(F, -2, 2, -3, 3, 50, 50)**

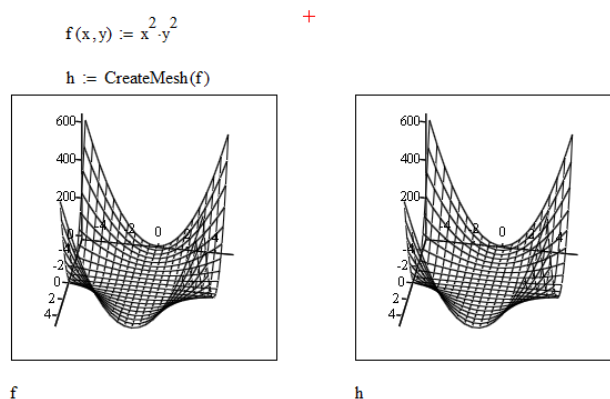


Рис. 3.47. Побудова зображення функції двох змінних

Для побудови кількох зображень одночасно слід ввести імена функцій чи матриць через кому.

Не обов'язково, щоб поверхні визначались однаковим способом. Можливо визначення однієї поверхні функцією, іншої – матрицею.

*Mathcad* має вбудовану функцію, яка призначена для перерахунку функції  $F(i, j)$ , в матрицю з  $m$  рядками та  $n$  стовпцями:

**matrix(m, n, F)** , де  $i = 0..m - 1$ ,  $j = 0..n - 1$

Функція **matrix** працює з **ORIGIN=0**, значення **ORIGIN=1** НЕ ПРАЦЮЄ.

Функція **matrix** має обмеження в значеннях аргументів функції  $F$  тільки натуральними числами.

### Параметричні тримірні графіки

Для складних поверхонь зручно використовувати їхні визначення в параметричному вигляді залежності координат функцією від двох змінних.

Для побудови параметричного 3D графіка слід ввести в дужках імена трьох функцій/матриць через кому або ім'я вектора, елементами якого є параметричні функції.

Система допускає опис функції в декартовій, циліндричній та сферичній системах координат. Для спрощення перерахунків можна використовувати вбудовані функції:

**xyz2pol (x, y)** – перерахунок декартових координат в полярні;

**pol2xyz (r, φ)** – перерахунок полярних координат в декартові;

**xyz2cyl (x, y, z)** – перерахунок декартових координат в циліндричні;

**cyl2xyz (r, φ, z)** – перерахунок циліндричних координат в декартові;

**xyz2sph (x, y, z)** – перерахунок декартових координат в сферичні;

**sph2xyz (x, φ, λ)** – перерахунок сферичних координат в декартові.

Результатом функції є вектор, який містить координати у відповідній системі координат. Для перерахування значень масиву слід застосовувати індексні чи векторизовані операції з векторами.

Функції перетворення можуть мати кутові координати в радіанах чи градусах. Для введення кутів в градусах слід додати службове слово **deg**. Для виведення кутів в градусах слід вписати **deg** в поле праворуч від числа.

Наприклад, для точки P з декартовими координатами x=2, y=8, z=5 сферичні координати можна перерахувати наступним чином:

$$\begin{array}{l}
 x := 2 \quad y := 8 \quad z := 5 \quad p := \begin{pmatrix} x \\ y \\ z \end{pmatrix} \\
 \hline
 xyz2sph(x, y, z) = \begin{pmatrix} 9.644 \\ 1.326 \\ 1.026 \end{pmatrix} \quad xyz2sph(p) = \begin{pmatrix} 9.644 \\ 1.326 \\ 1.026 \end{pmatrix} \quad xyz2sph(p) = \begin{pmatrix} 552.54 \\ 75.964 \\ 58.77 \end{pmatrix} \text{deg}
 \end{array}$$

Якщо чисельні значення не потрібні, то перерахунок систем координат можна задавати на вкладинці **Quick Plot Data** вікна форматування 3D графіків.

Наприклад, сферична поверхня з центром в точці початку координат та радіусом r в декартових координатах параметрично описується як

$$\mathbf{X}(u, v) = r \cdot \sin(u) \cos(v) \quad \mathbf{Y}(u, v) = r \cdot \sin(u) \sin(v)$$

$$\mathbf{Z}(u, v) = r \cdot \cos(u) .$$

Зображення поверхні через параметризовані функції та вектор в декартових координатах наведено на рис. 3.48. Параметри графіка при такому визначенні даних можна змінити ТІЛЬКИ у вікні редагування.

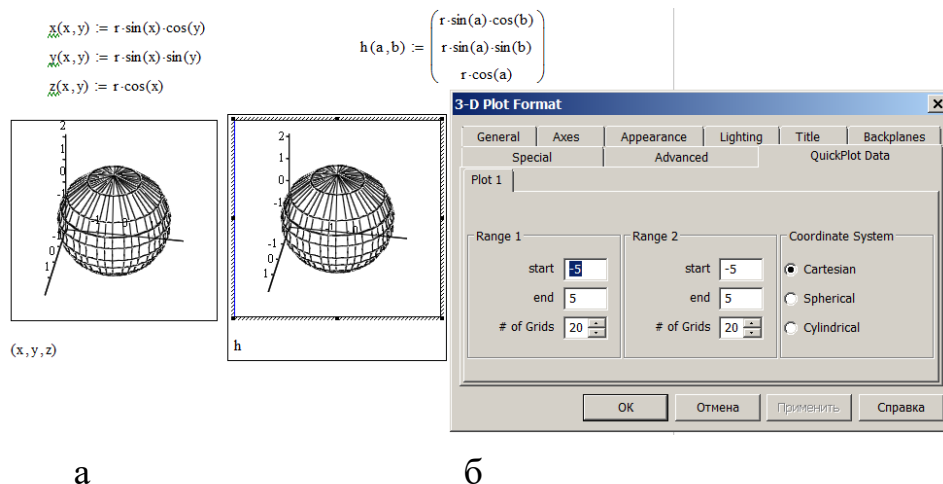


Рис.3.48. Зображення сфери: а – параметризованими функціями в декартових координатах; б – вектором в декартових координатах

Недоліком цього способу можна вважати те, що точно визначити параметри графіка у вікні, наприклад, через формульні залежності, неможливо, для визначення потрібно робити зайві дії, щоб дістатися до полів редагування. Програмний контроль забезпечує саме функція **CreateMesh**.

Графік поверхні із використанням параметризації та функції **CreateMesh** показано на рис. 3.49. При такому визначенні даних вікно редагування діапазонів стає недоступним.

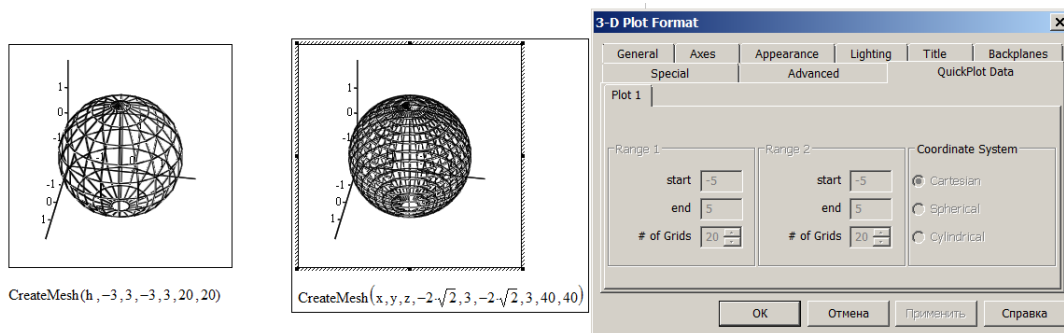


Рис. 3.49. Зображення сфери із функцією **CreateMesh** в декартових координатах

Опис сфери в сферичних координатах суттєво спрощується. Поверхня визначається радіусом  $r$ , та двома кутами: місця, який вимірюється від



площини  $XOY$  до точки та має діапазон  $-\pi \dots \pi$ , азимута, який вимірюється в площині  $XOY$  в діапазоні  $0 \dots 2\pi$ .

Для отримання зображення із функціями в сферичних координатах слід в дужках ввести через кому імена параметричних функцій: кута місця, азимута, радіуса та на вкладниці **QuickPlot Data** обрати систему координат **Spherical** (рис. 3.50).

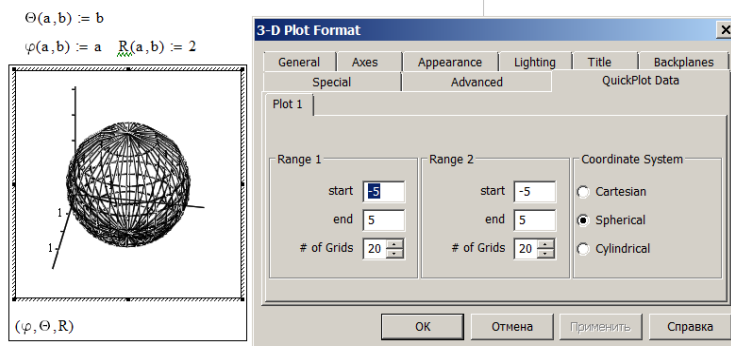


Рис. 3.501 Зображення сфери в сферичних координатах

Як і у випадку декартових координат, можна застосувати функцію **CreateMesh**. Результат застосування функції наведено на рис. 3.52. Сторінка визначення діапазонів та систем координат графіка і в цьому випадку стає недоступною. Особливістю є різний порядок розташування елементів в векторах. Для безпосереднього застосування він співпадає з порядком розташування імен параметричних функцій: кут місця, азимут, радіус. Виклик функції потребує іншого розташування імен: радіус, кут місця, азимут.

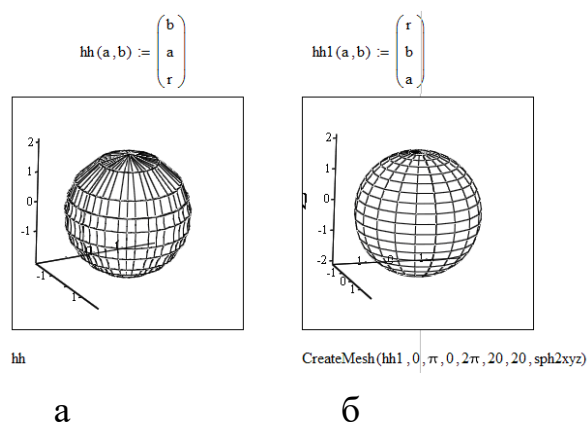


Рис. 3.51. Зображення сфери в сферичних координатах: а – через вектор; б – через функцію **CreateMesh**

## Контурні графіки

Просторове зображення дає загальну уяву про функцію або поверхню. Для кількісного аналізу можна використати графіки ізоліній **Contour Plot** з сторінки **General** вікна редагування графіків. Цей графік зображує поверхню ортогонально площині  $XOY$  уздовж осі  $Z$ . При цьому на зображенні виводяться лінії визначених значень функції – ізолінії та кількісні позначки (рис. 3.52).

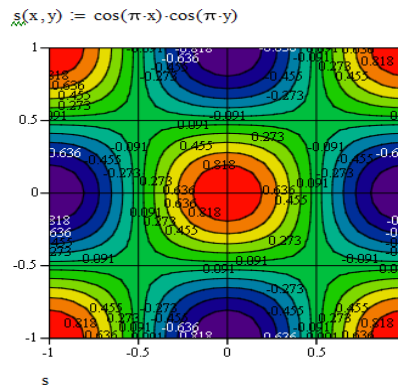


Рис. 3.52. Контурний графік

## Графіки просторових кривих

Для побудови просторових кривих застосовується графіки типу *Data/Scatter Plot*. Для опису кривої використовується параметрична форма. Для побудови зображення необхідно підготувати три вектори чи функції із визначенням аргументів  $x$ ,  $y$ ,  $z$ . Розмір векторів визначає кількість точок графіка в самому описі. При використанні функцій кількість точок за замовчанням 20, зміна кількості проводиться на вкладниці **QuickPlot Data** вікна редагування графіка тільки вручну.

Для формування векторів призначена вбудована функція

**CreateSpace(F, min, max, num, fmap),**

де **F** – вектор, елементами якого є три вирази розрахунку координат  $x$ ,  $y$ ,  $z$  в параметричному вигляді, **min**, **max** – початкове та кінцеве значення аргументу, **num** – кількість точок, **fmap** – необов'язкова функція перерахунку системи координат, якщо в векторі **F** застосована не декартова система координат.

За замовчанням  $\min=-5$ ,  $\max=5$ ,  $N=20$ .

Наприклад, просторова спіраль описується параметричними функціями у вигляді:

$$X(t) = r \cdot \cos(t) \quad Y(t) = r \cdot \sin(t) \quad Z(t) = t$$

Для побудови графіка слід:

1. Визначити функції користувача  $X(t)$ ,  $Y(t)$ ,  $Z(t)$ . Параметр  $k$  позначає кількість витків спіралі.

2. В маркер тримірного графіка ввести в дужках через кому імена функцій. За замовчанням буде зображено графік поверхні (рис. 3.53).

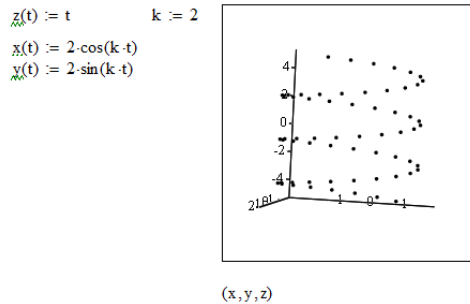


Рис. 3.53. Зображення спіралі до редагування

3. За замовчанням графік зображується у вигляді точок. На вкладинці **Appearance** обрати в групі **Line options** пункт **Line**. Графік набуває вид, який представлено на рис. 3.54.

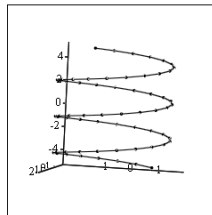


Рис. 3.54. Графік просторової спіралі

Аналогічного результату можна досягти з використанням функції - вектора. Технологія побудови зображення відрізняється тим, що в маркер ставиться тільки одне ім'я вектора. На рис. 3.55 наведено графік, для якого на сторінці **Appearance** в групі **Point options** додатково визначено опорні точки розміром 3 червоного кольору .

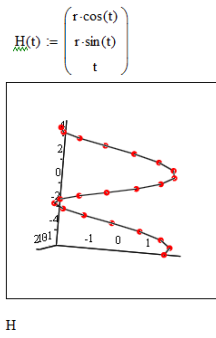


Рис. 3.55. Графік спіралі з використанням вектора

Програмне керування кількістю точок графіка забезпечують вихідні дані у вигляді векторів або функція **CreateSpace**. В такому способі аргументи визначаються явно (рис. 3.56).

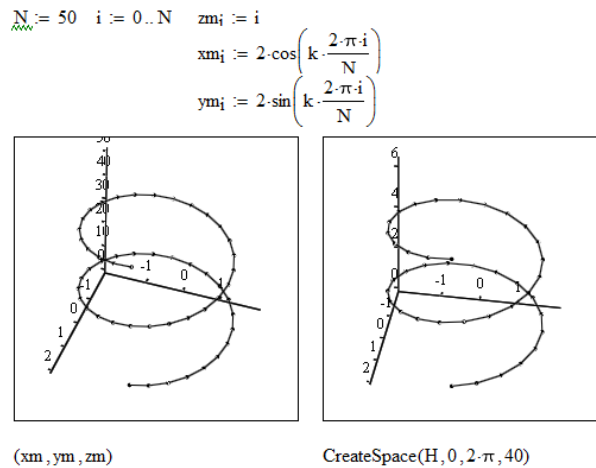


Рис. 3.56. Застосування векторів та функції **CreateSpace**

### Графік векторного поля

Графік векторного поля активується пунктом **Vector Field Plot** діалогового вікна тримірних графіків. Зображення являє собою карту векторів. Кожен вектор починається у вузлі сітки та визначається локальними координатами відносно вузла сітки x, y. Даними для цього виду графіків є матриці, кожен елемент яких містить дві координати x, y вектора.

Зображення виводиться для транспонованої матриці. Рядки вихідної матриці зображуються стовпцями. Для абсцис додатній напрямок праворуч, від'ємний – ліворуч, для ординат додатній напрямок вгору, від'ємний – вниз.

Вихідна матриця може готуватися кількома способами.

Спосіб 1. Кожен елемент матриці вважається комплексним числом. Дійсна частина відповідає координаті  $x$ , уявна – координаті  $y$  вектора.

Спосіб 2. Дані вводяться у дві матриці. Перша матриця містить координати  $x$ , друга –  $y$  векторів. Матриці поєднуються в комплексну так, щоб перша відповідала дійсній частині, друга – уявній. Можливо вивести зображення без поєднання як параметричне з двох вихідних матриць.

Наприклад, для значень координат  $x = \cos(xy)$   $y = \sin(xy)$  побудова карти способом 1 може мати вигляд, зображений на рис. 3.57 а, для способу 2 з однією матрицею – на рис. 3.57 б, для способу 2 параметрично – на рис. 3.57 в.

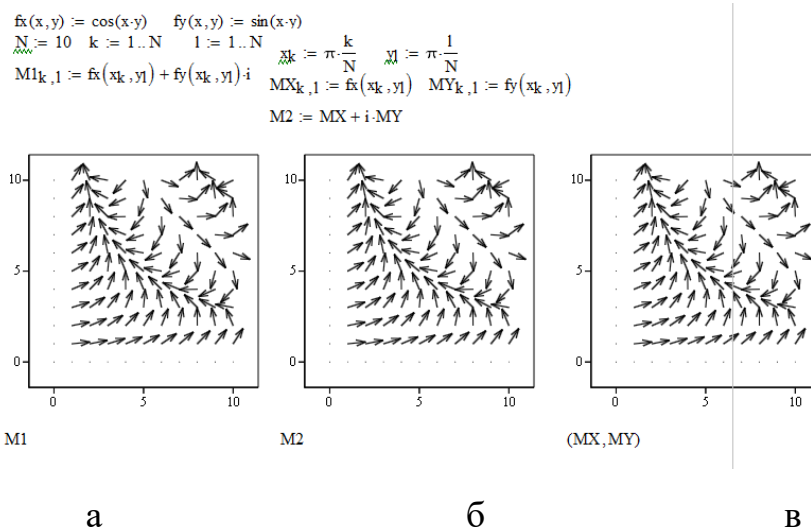


Рис. 3.57. Векторне поле: а – спосіб 1; б – спосіб 2; в – спосіб 2 параметрично

### Зображення з «тінню»

Для більшої інформативності використовуються зображення функції з «тінню». В якості тіні можна застосувати контурний графік або фон з сіткою. Для контурного графіка в якості тіні ім'я функції вводиться двічі (рис. 3.58).

$$f(x, y) := x^2 + y^2$$

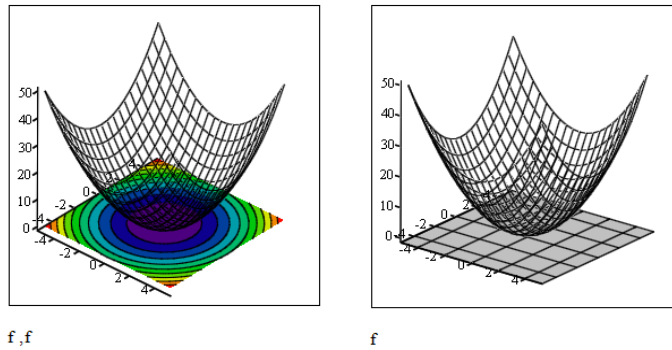


Рис. 3.58. Зображення поверхні з «тінню».

*MathCAD Prime* має дещо скорочені типи тримірних графіків та можливості візуальних ефектів з одночасним спрощенням форматування графіків.

Система спроможна відображати поверхневі графіки **Surface Plot** (рис. 3.59 а), контурні графіки **Contour Plot** (рис. 3.59 б). Поверхневі графіки можуть бути зображені у вигляді 3D точкової діаграми (рис. 3.59 в),

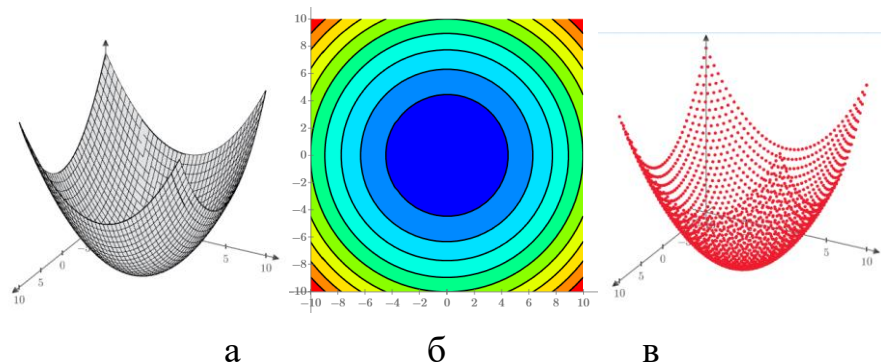


Рис. 3.59. Типи тримірних графіків: а – поверхні; б – контурний; в – точковий

Створення графічної області 3D графіків здійснюється клавішами **CTRL+3**, пунктом меню **0** стрічки.

Результатом є відображення на екрані шаблону 3D графіки з єдиним маркером (рис. 3.61) позначками дій редагування графіка ліворуч та вертикальною шкалою активної осі графіка праворуч.

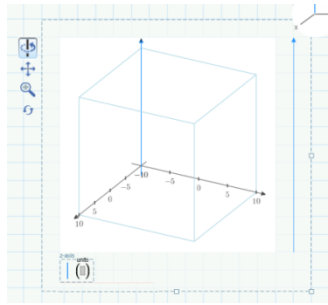


Рис. 3. 60. Шаблон тримірних графіків

На місці маркера графіка для побудови 3D графіка необхідно ввести ім'я матриці, для якої передбачається відображення у вузлах, або ім'я наперед визначеної функції користувача від двох змінних.

Для побудови кількох графіків на одному графічному полі слід використати кнопку **Plots-Traces-Add Trace** стрічки.



Безпосередньо на графічному полі можна повернути, зсунути зображення та змінити його масштаб.

Активна ось графіка виділяється синім кольором. На вертикальній шкалі праворуч від графіка можна визначити діапазон осі та мітки на ній. Для зміни значення максимального/мінімального значення осі слід клікнути на відповідному значенні шкали та ввести потрібне значення. Кількість міток на осі визначається автоматично за двома крайніми значеннями шкали. Для зміни кількості міток на осі слід змінити попереднє перед останнім або наступне після першого значення шкали.

Вибір активної осі проводиться кліком на позначці трикутника осей в правому верхньому куті поля графіки.

*За замовчанням аргументи функції вважаються заданими в діапазоні від -10 до 10 в 41-й точці.*

*Дозволяється введення замість імені функції користувача розрахункового виразу.*

В разі створення графіка функції, яка явно залежить від змінних  $x, y$ , кількість точок графіка та діапазони аргументів можна задати ранжованими змінними за умови введення імені функції та аргументів. Діапазон також

визначається значеннями осей, кількість точок – полем *Plots-Traces-Number of Points* стрічки.

В разі створення графіка матриці кількість точок графіка та діапазони аргументів є незмінними та визначені самою матрицею.

Створення графічної області контурних графіків здійснюється клавішами *CTRL+5*, пунктом меню **Plot-Insert Plot-Contour Plot** стрічки. Результатом є відображення на екрані шаблону 3D графіки з єдиним маркером (рис. 3. 61)

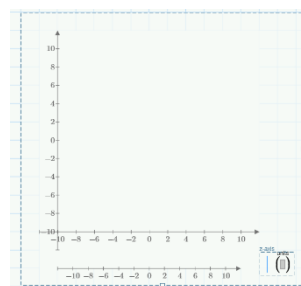


Рис. 3.61. Шаблон контурних графіків

В маркер поля для побудови контурного графіка необхідно ввести ім'я матриці, для якої передбачається відображення у вузлах, або ім'я наперед визначеної функції користувача від двох змінних.

На контурний графік (рис. 3.62) можна вивести контурні лінії для визначених значень функції – ізолінії визначеним візерунком **Line Style**, кольором **Trace Color** та товщиною **Trace Thickness**, нанести на контурні лінії чисельні значення функції **Contour Values** з групи **Style** стрічки **Plots**. Там же можна визначити кольорову палітру фарбування графіка **Color Scheme**. Припустими є сім стандартних схем: без фарбування *None*, веселка *Rainbow*, градації сірого *Grayscale*, освітлені градації сірого *Gamma*, градації червоного *Red*, синього *Blue*, зеленого *Green*, топографічна *Topographical*.

Під контурним графіком розташована шкала кольорів. За замовчанням шкала має 10 градацій від мінімального значення **MIN** до максимального **MAX** із сталим кроком. Чисельні значення нанесено на шкалу. Крок значень для зміни кольорів визначається як різниця між двома сусідніми значеннями шкали.



Змінити діапазон аргументів контурного графіка, значення шкали кольорів можна редагуванням відповідних значень. Для модифікації значення слід навести курсор у вибране місце. Доступні для редагування значення змінюють свій розмір. Режим редагування активується клацанням мишею над обраним місцем. Після зміни будь-якого значення контурний графік та шкала кольорів перемальовуються автоматично.

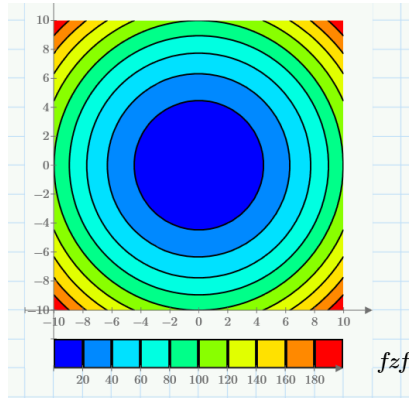


Рис. 3.62. Контурний графік

### 3.4.2. Приклади та завдання для самостійного виконання

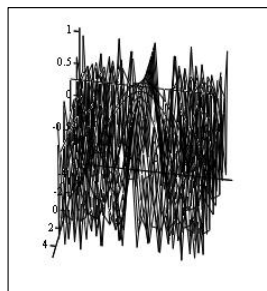
*Приклад 3.4.1.* Побудувати графік функції  $z(x,y)=\cos(x \cdot y)$  в діапазоні аргументів  $-2 \dots 2$  із кроком 0.12 стандартним способом та з функцією **CreateMesh**.

#### РОЗВ'ЯЗАННЯ

1. Розрахувати кількість інтервалів аргументу  $N = \frac{2 - (-2)}{0.12} = 33$

2. Визначити функцію користувача  $\mathbf{z(x,y)} := \mathbf{\cos(x \cdot y)}$ .

3. В маркер тримірного графіка вставити ім'я функції. Зображення за замовчанням (рис. 3.64) не відповідає завданню як за виглядом, так і за діапазонами аргументів та кількістю інтервалів на діапазоні.



z

Рис. 3.63. Первинне зображення

4. Змінити тип фарбування на вкладниці **Appearance** на *Fill Surface* та *Colormap*, визначити на вкладниці **Quick Plot** діапазони обох аргументів [ -2 2] та кількість ліній 33. Зображення набуває потрібного вигляду (рис. 3.64 а).

5. В маркер нового поля графіка ввести функцію **CreateMesh** з визначеними аргументами. Змінити тип фарбування на вкладниці **Appearance** на *Fill Surface* та *Colormap*. Зображення стає аналогічним п.4 (рис. 3.64 б)

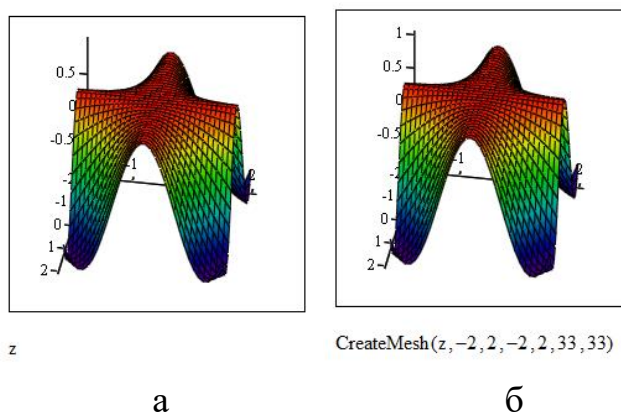


Рис.3.64. Зображення: а – після ручного редагування; б – з функцією **CreateMesh**

*Приклад 3.4.2.* Накреслити поверхневий матричний, контурний та тримірний точковий графіки функції  $f(x,y)=0.5-x^2-y^2$  з кроком обох аргументів 0.25 в межах  $-1 < x < 1$ ,  $-1 < y < 1$ .

### РОЗВ'ЯЗАННЯ

1. Визначити розмір матриці  $N = \frac{1 - (-1)}{0.25} = 8$ .

2. Визначити елементи матриці через індекс у вигляді ранжованої змінної та крок 0.12

$$N := 8 \quad \text{ORIGIN} := \frac{-N}{2} \quad \text{step} := 0.25 \quad +$$

$$i := \frac{-N}{2} .. \frac{N}{2} \quad j := \frac{-N}{2} .. \frac{N}{2} \quad x_i := i \cdot \text{step} \quad y_j := j \cdot \text{step}$$

$$M_{i,j} := \left[ 0.5 - (x_i)^2 \right] - (y_j)^2$$

3. Графік поверхні. В поле маркера тримірного графіка вставити ім'я матриці. Зображення наведено на рис. 3.65 а.

5. Точкове зображення. В поле маркера тримірного графіка вставити ім'я матриці. На вкладниці **General** встановити тип *Data Points*, на вкладинці **Appearance** встановити *Symbol Dot* розміром 2 та кольором *Red*. Зображення наведено на рис. 3.66 б.

4. Контурний графік. В поле маркера тримірного графіка вставити ім'я матриці. На вкладниці **General** встановити тип *Contour Plot*, на вкладинці **Special** встановити фарбування *Fill*, тип *Numbered* з кількістю ліній 5. На вкладниці **Advanced** обрати *Colormap Grayscale*. Зображення наведено на рис. 3.65 в.

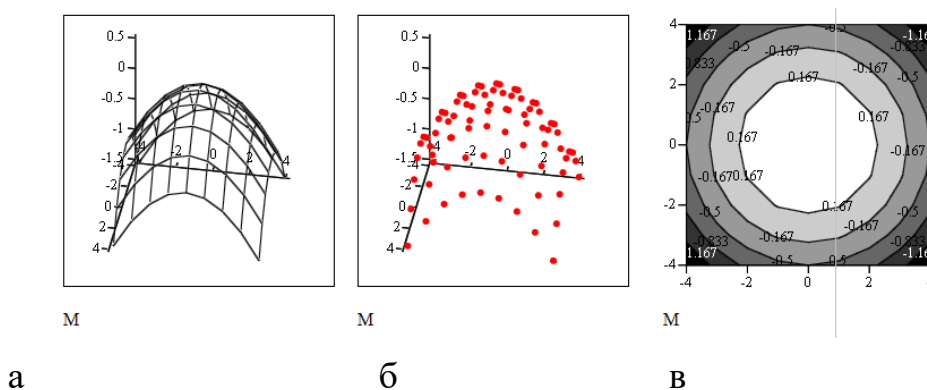


Рис. 3.65. Зображення прикладу 3.4.2: а – матричний графік поверхні; б – точковий графік; в – контурний графік

*Приклад 3.4.3. Параметричне зображення.*

Рівняння торичної поверхні записується як:

$$y(a,b)=(R_{\max}+R_{\min}\cdot\cos(a))\cdot\sin(b)$$

$$x(a,b)=(R_{\max}+R_{\min}\cdot\cos(a))\cdot\cos(b)$$

$$z(a,b)=R_{\min}\cdot\sin(a)$$

Діапазон зміни кута  $a$  –  $[0 .. \pi]$   $b$  –  $[0.. 2\pi]$ .

Намалювати поверхню тора із секторним вирізом  $\pi/8$  по куту  $b$  параметричним способом та функцією **CreateMesh**. Великий радіус тора 5, малий - 2. Число точок 20.

### РОЗВ'ЯЗАННЯ

1. Визначити вихідні дані **Rmax:=5 Rmin:=2 N:=20**.

2. Параметричний спосіб.

- визначити матриці X, Y, Z. Крок по куту становить  $\frac{2\pi}{20} = \frac{\pi}{10}$ , тому точно виконати умови завдання неможливо. Для цього способу виріз становитиме один крок

$$i := 0..N \quad a_i := i \cdot \frac{\pi}{N} \quad j := 0..N-1 \quad b_j := 2 \cdot j \cdot \frac{\pi}{N}$$

$$X_{i,j} := (R_{\max} + R_{\min} \cdot \cos(a_i)) \cdot \cos(b_j)$$

$$Y_{i,j} := (R_{\max} + R_{\min} \cdot \cos(a_i)) \cdot \sin(b_j)$$

$$Z_{i,j} := R_{\min} \cdot \sin(a_i)$$

- в полі тримірного графіка ввести в дужках назви матриць X, Y, Z. Отримане зображення наведено на рис. 3.66 а.

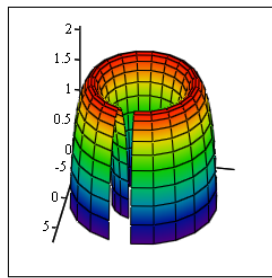
3. Спосіб вбудованої функції.

- визначити векторну функцію користувача  $h(a,b)$

$$h(a,b) := \begin{bmatrix} (R_{\max} + R_{\min} \cdot \cos(a)) \cdot \cos(b) \\ (R_{\max} + R_{\min} \cdot \cos(a)) \cdot \sin(b) \\ R_{\min} \cdot \sin(a) \end{bmatrix}$$

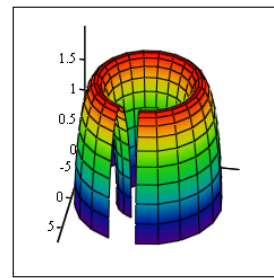
- в полі тримірного графіка ввести функцію **CreateMesh(h, 0,  $\pi$ , 0,  $2\pi - \pi/8$ , 20, 20)**. Отримане зображення наведено на рис. 3.66 б.

Для обох способів графік функції має однаковий вигляд.



(X, Y, Z)

а



CreateMesh(h, 0, π, 0, 2π - π/8, 20, 20)

б

Рис.3.66 Зображення торичної поверхні: а – параметричне; б – функцією **CreateMesh**

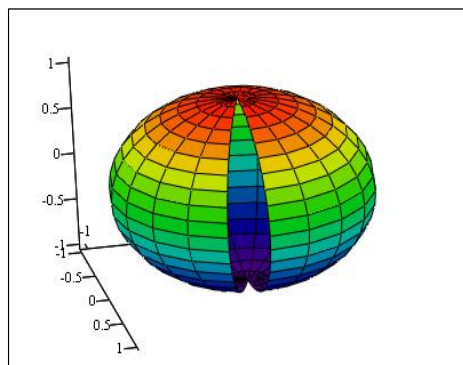
*Завдання 3.4.1.* Побудувати двомірний, тримірний та контурний графіки, що ілюструють «кільця Ньютона» згідно наступного виразу:

$$f(r) = I_0 \cdot \cos^2\left(\frac{2 \cdot \pi}{\lambda} \cdot \delta(r)\right), \text{ де}$$

$I_0=1, \lambda=0.5 \cdot 10^{-3}, R=5000, 0 < r < 4$ , крок 0.02.

$$\delta(r) = \sqrt{R^2 + r^2} - R.$$

*Завдання 3.4.2.* Намалювати зображення сфери із секторним вирізом (рис. 3.67) параметрично та функцією **CreateMesh**.



(X, Y, Z)

Рис. 3.67. Сфера завдання 3.4.2

*Завдання 3.4.3.* На інтервалі -1 ... 1 зобразити каркасно, з кольоровим фарбуванням, лініями рівня поверхню

$$f(x, y) = (\sin(x^2) + \cos(y^2))^{xy}.$$

*Завдання 3.4.4.* Зобразити поверхню, яка визначена параметрично  
 $x(u,v)=\cos(u)\cdot\sin(v)$   $y(u,v)=\sin(u)\cdot\sin(v)$   $z(u,v)=u\cdot v$   $u,v\in[0,3]$ .

*Завдання 3.4.5.* Намалювати гвинтову лінію  $x(t)=t\cdot\cos(t)$   $y(t)=t\cdot\sin(t)$   $z=t$ .

*Завдання 3.4.6.* Намалювати векторне поле

$$x(x,y)=\cos(3\pi x) \quad y(x,y)=\sin(3\pi y).$$

*Завдання 3.4.7.* Намалювати циліндричну поверхню, яка визначена в циліндричних координатах

$$R(a,b)=2 \quad \varphi(a,b)=a \quad q(a,b)=b.$$

## **Практикум 3.5. Розрахунки з розмірностями**

### **3.5.1. Теоретичні положення**

В *Mathcad*, на відміну від більшості СКМ, передбачена можливість проведення технічних, фізичних розрахунків з розмірними величинами. Змінну, значенням якої є величина з розмірністю, називають розмірною змінною.

В *Mathcad* забезпечує розрахунки в таких системах розмірностей:

- система СІ (SI (International));
- система МКС (MKS, метр, кілограм, секунда);
- система СГС (CGS, сантиметр-грам-секунда);
- американська система одиниць (U.S.);
- система користувача (Custom).

Примітка. Останню можливість не можна вважати системою користувача, бо це є тільки можливість скомбінувати одиниці розмірності з вище наведених систем. Додати власні або змінити мову таким чином неможливо.

Фізичні розрахунки проводяться з застосуванням базових одиниць виміру: L – length (довжина), M – mass (маса), T – time (час), Q – charge (заряд). K – temperature (температура), I – luminosity (сила світла), S – substance (одиниця речовини).

Наприклад, в системі СІ основні одиниці виміру мають наступний вигляд: **m (length, meter), kg (mass, kilogramm), s (time, second) q (charge, coulomb), K (temperature, kelvin), cd (luminosity, candella).**

Через основні одиниці виміру визначаються інші додаткові одиниці. Наприклад, **cm:=0.01 ·m, mm:=0.001 ·m, km:=1000 ·m, tonne:=103 ·kg, hr:=3600 ·sec, min:=60 ·s, A:=coul ·sec<sup>-1</sup>.**

Незалежно від використаної системи розмірностей, результат операції в *Mathcad* переважно виражається в фундаментальних одиницях: довжина, маса, час, заряд, одиниця речовини, температура, сила світла. Конкретні розмірності виставляє користувач вручну.

Частково зробити приведення розмірностей система спробує, якщо на сторінці (рис. 3.68) пункту меню **Format-Result Unit Display** обрано опцію **Simplify units when possible**.

Сторінка містить опції: *Format units*, *Simplify units when possible*, *Show unit exponents as a fraction*.

Якщо встановлено опцію *Format units*, вираз для розмірної частини значення відображається у вигляді дроби. Опція *Simplify units when possible* встановлюється для автоматичного приведення одиниць вимірювання. Опція *Show unit exponents as a fraction* може бути встановлена тільки для всього документа в цілому і забезпечує відображення одиниць розмірностей у вигляді раціональних дробів.

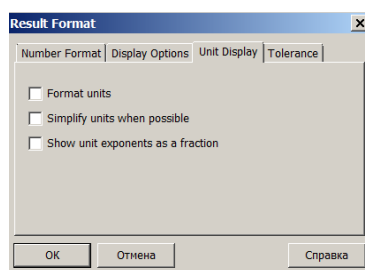


Рис. 3.68. Сторінка форматування розмірностей

Якщо діалогове вікно **Result Format** відкривалося подвійним клацанням миші на результаті, то формат застосовується тільки до цього результату. Для глобального визначення вигляду одиниці вимірювання у

всьому документі діалогове вікно **Result Format** слід відкривати пунктом меню **Format- Result**.

Визначення системи розмірностей проводиться на сторінці **Unit System** (рис. 3.69 а) пункту, меню **Tools-Worksheet Options**. Для вимкнення розмірностей слід обрати пункт *None*. Для *MathCAD Prime* визначення системи проводиться зі списку **Math-Units-Unit System** стрічки (рис. 3.69 б).

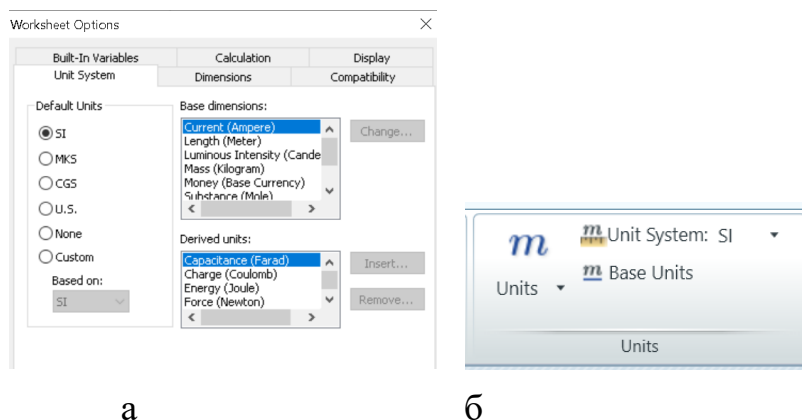



Рис. 3.69. Системи розмірностей: а – *MathCAD*; б – *MathCAD Prime*

Для використання розмірної змінної слід ввести її ім'я, значення та знак множення. В поле маркеру після знаку множення вводиться назва розмірності. За замовчанням розмірності вводяться АНГЛІЙСЬКОЮ МОВОЮ.

Для введення розмірностей можна використати діалогове вікно **Insert Unit** через поле **Insert-Unit** меню або ще простіше – кнопкою  меню **Standard** (рис. 3.70).

Для скорочення часу для пошуку потрібної розмірності всі розмірності, які є в системі, розбиті по категоріях. У вікні є два списки: *Dimension* та *Unit*. Список *Dimension* представляє категорії розмірностей. Список *Unit* містить конкретні розмірності відповідної категорії (табл. 3.7).



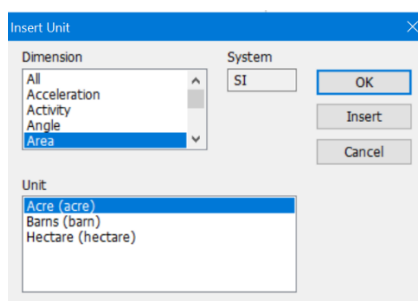


Рис. 3.70. Діалогове вікно **Insert Unit**

Таблиця 3.7. Типи розмірних величин Mathcad

Категорія	Опис
All	Всі доступні розмірності
Acceleration	Одиниці розмірності присквидшення
Activity	Одиниці розмірності радіоактивності
Angle	Одиниці розмірності кутів
Area	Одиниці розмірності площ
Capacitance	Одиниці розмірності електричної ємності
Catalytic Activity	Одиниці розмірності каталітичної активності
Charge	Одиниці розмірності електричного заряду
Conductance	Одиниці розмірності електричної провідності
Current	Одиниці розмірності сили електричного струму
Dose	Одиниці розмірності доза випромінювання
Energy	Одиниці розмірності енергії
Flow Rate	Одиниці розмірності потоку рідини
Force	Одиниці розмірності сили
Force Density	Одиниці розмірності об'ємної щільності сили

Force per Length	Одиниці розмірності лінійна щільності сили
Frequency	Одиниці розмірності частоти
Illuminosity	Одиниці розмірності освітленості
Inductance	Одиниці розмірності індуктивності
Length	Одиниці розмірності довжини

Таблиця 3.7. Продовження

Категорія	Опис
Potential	Одиниці розмірності електричного потенціалу
Power	Одиниці розмірності потужності
Pressure	Одиниці розмірності тиску
Resistance	Одиниці розмірності електричного опору
Substance	Одиниці розмірності кількості речовини
Temperature	Одиниці розмірності температури
Time	Одиниці розмірності часу
Velocity	Одиниці розмірності швидкості
Viscosity, dynamic	Одиниці розмірності динамічної в'язкості
Volume	Одиниці розмірності об'єму
Magnetic Flux Density	Одиниці щільності магнітного потоку
Mass	Одиниці розмірності маси
Permeability	Одиниці розмірності магнітної проникненості

Permittivity	Одиниці розмірності електричної проникності
Luminous Intensity	Одиниці розмірності яскравості світла

Користувач має змогу ввести свої додаткові одиниці по аналогії з вбудованими.

Наприклад, саме так можна перевести розмірності на іншу мову або створити власні позасистемні одиниці:

**см:=0.01·m мм:=mm В:=V година:=3600·sec Вт:=watt  
квартира:=60·м2 дім:=100·квартира людина:=1.**

При введенні розмірностей користувача, які не пов'язані із вбудованими розмірностями, *Mathcad* не слідкує за правильністю розмірностей. Вся відповідальність за результати покладається на користувача, який повинен сам вписувати у відповіді правильну розмірність.

Розмірність може призначатися змінним, ранжованим змінним, масивам, функціям користувача.

Розмірність для ранжованої змінної вводиться для кожного зі значень, розмірність масивів вводиться як і для чисел. Розмірність всіх елементів масиву має бути однаковою.

Наприклад, **a:=10сек, 11сек . . 20сек**

$$v = \begin{pmatrix} 20 \\ 40 \\ 60 \end{pmatrix} \text{мм} \quad v = \begin{pmatrix} 0.02 \\ 0.04 \\ 0.06 \end{pmatrix} \text{метр} \quad v = \begin{pmatrix} 2 \\ 4 \\ 6 \end{pmatrix} \text{см}$$

Графіки відображаються в стандартних розмірностях. В разі використання умовного оператора результати можуть бути тільки безрозмірними.

### **Розмірність в підпрограмах**

Для перевірки розмірностей в версіях з 13-ї слід ввести ім'я функції без параметрів. Результат має вигляд, подібний наступному

$$vp(t,v) := t \cdot v \quad vp = f(\text{any1}, \text{any2}) \rightarrow \text{any1} \cdot \text{any2}$$

Результат містить символічну позначку функції  $f$  з переліком аргументів **any** та стрілку. Після стрілки розташовується пояснення. Якщо параметр може мати розмірність, то виводиться його ім'я та ступінь розмірності, якщо ні - то виводиться службове слово **unitless**. В *MathCAD Prime* перевірка розмірності підпрограм-функцій не реалізована.

Розплатою за введenu перевірку розмірностей стало те, що в версіях системи від 13-ї деякі вбудовані функції не працюють з розмірностями, а саме: функції інтерполяції, регресії, розв'язання рівнянь, обчислення логарифмів, створення сітки для тримірних графіків.

Розмірності в підпрограмах можна застосовувати тільки коли всі результати мають однакову розмірність. Наприклад, в попередніх версіях системи існувала можливість виконання підпрограми, яка розраховувала периметр та площу геометричної фігури. В останніх версіях така підпрограма неможлива, потрібно окремо розраховувати площу та периметр.

Для введення одиниць розмірности в *MathCAD Prime* використовується панель розмірностей **Math-Units-Base Units** стрічки (рис. 3.71). Панель розмірностей **Base Units** містить позначки вбудованих одиниць розмірности, які згруповані по категоріях.

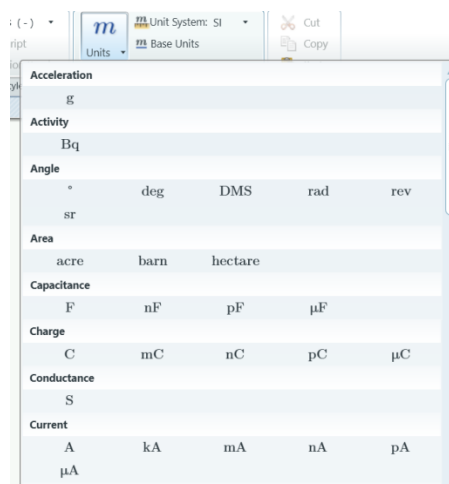


Рис. 3.71. Панель розмірностей

### 3.5.2. Приклади та завдання для самостійного виконання

*Приклад 3.5.1.* Визначити за законом Ньютона силу тіла масою 1 кг, яке має прискорення  $2 \text{ м/с}^2$ .

#### РОЗВ'ЯЗАННЯ

1. Визначити змінну для маси тіла, після чисельного значення ввести позначку множення  $mas := 1 \cdot$ , на місце маркеру ввести розмірність маси **kg**.

2. Ввести змінну прискорення  $a$ , після чисельного значення поставити позначку множення  $mas := 1 \cdot kg$  |  $a := 2 \cdot$  на місце маркеру ввести розмірність прискорення **m/s<sup>2</sup>**.

3. Записати формулу закону Ньютона та поставити знак виведення результату «="". Отримуємо результат обчислення. Розмірність виведена в фундаментальних одиницях **m·kg·s<sup>-2</sup>**:

$$mas := 1 \cdot kg \quad a := 2 \cdot \frac{m}{s^2}$$

$$mas \cdot a = 2 \cdot m \cdot kg \cdot s^{-2}$$

4. На місці маркеру ввести **newton**       $mas \cdot a = 2 \text{ newton}$

$$mas \cdot a = 2 \cdot N$$

*Приклад 3.5.2.* Визначити щільність та питому вагу сталі, якщо 2 см<sup>3</sup> її об'єму мають масу 15.7 г.

### РОЗВ'ЯЗАННЯ

Щільність визначається діленням об'єму речовини на масу. Питома вага – множенням прискоренням вільного падіння на щільність.

1. Ввести вихідні значення для об'єму в кубічних сантиметрах, маси в грамах

$$V := 2 \cdot \text{cm}^3 \quad mas := 15.7 \cdot \text{gm}$$

2. Розраховувати щільність та питому вагу

$$\rho := \frac{mas}{V} = 7.85 \times 10^3 \frac{kg}{m^3}$$

$$uv := \rho \cdot g = 7.698 \times 10^4 \frac{kg}{m^2 \cdot s^2}$$

Зверніть увагу, на незвичне позначення грамів та на той факт, що стала вільного падіння є вбудованою в систему.

*Приклад 3.5.3.* Визначити енергію вантажу за законом Гука  $E = k \cdot \frac{l_0^2}{2}$ ,

що підвішений на пружині при початковому зсуві  $l_0=8$  см, якщо відомо, що сила  $F=2$  кГс викликає подовження пружини на  $L=10$  мм. Коефіцієнт Гука

$$k = \frac{F}{L}$$

#### РОЗВ'ЯЗАННЯ

$$F_0 := 2 \cdot kgf \quad L_0 := 8 \cdot cm \quad Len := 10 \cdot cm \quad - \text{визначення вихідних}$$

даних сили в одиницях «кілограм сили», подовження в одиницях «сантиметр», початкового зсуву в одиницях «сантиметр». Для змінної сили, подовження застосовано імена  $F_0$ ,  $Len$ , бо в системі назва  $F$  вже використано для одиниці виміру «Фарада»,  $L$  – одиниця виміру «літр».

$$k := \frac{F_0}{Len} = 196.133 \frac{kg}{s^2} \quad - \text{розрахунок коефіцієнта Гука.}$$

$$E := k \cdot \frac{L_0^2}{2} = 0.628 m^2 \cdot kg \cdot s^{-2} \quad E = 0.628 J \quad - \text{обчислення енергії.}$$

Перший результат виведено в базових одиницях. Другий – в Джоулях за рахунок увімкнення режиму спрощення розмірностей.

*Приклад 3.5.4.* Визначити променевий потік від абсолютно чорного тіла у вигляді диску діаметром 1 см в діапазоні  $\lambda$  (1...3) мкм для температури  $T=300K$  інтегруванням формули Планка.

#### РОЗВ'ЯЗАННЯ

Формула Планка визначає спектральну щільність світності для визначеної температури випромінювача

$$M(\lambda, T) = \frac{C_1}{\lambda^5 e^{\left(\frac{C_2}{\lambda T} - 1\right)}} \left[ \frac{W}{cm^2 mkm} \right]$$

де  $C_1 = 37400 \text{ Вт} \cdot \text{мкм}^4 \cdot \text{см}^{-2}$ ,  $C_2 = 14400 \text{ мкм} \cdot \text{К}$  – сталі формули Планка,  $\lambda$  – довжина хвилі випромінювання,  $T$  – температура.

Потік випромінювання визначається як світність, помножена на площу випромінювача:

$$F = M \cdot A, \text{ де } A = \frac{\pi d^2}{4} \text{ – площа випромінювача.}$$

1. Визначити вихідні дані: діаметр  $d$ , довжини хвилі початку  $\lambda_1$  та кінця діапазону  $\lambda_2$ , температуру  $T$

$$d := 1 \cdot \text{cm} \quad \lambda_1 := 1 \text{ micron} \quad \lambda_2 := 2 \cdot \text{micron} \quad T := 300 \cdot \text{K}$$

2. Визначити сталі формули Планка

$$C_1 := 37400 \cdot \text{W} \cdot \text{cm}^{-2} \cdot \text{micron}^4 \quad C_2 := 14400 \cdot \text{micron} \cdot \text{K}$$

3. Визначити розрахункову формулу Планка як функцію користувача

$$M(\alpha, \beta, t) := \int_{\alpha}^{\beta} \frac{C_1}{\lambda^5 \cdot \exp\left(\frac{C_2}{\lambda \cdot T} - 1\right)} d\lambda$$

4. Розрахувати площу випромінювача

$$A := \frac{\pi \cdot d^2}{4} = 7.854 \times 10^{-5} \text{ m}^2$$

5. Розрахувати променевий потік

$$\text{Flux} := M(\lambda_1, \lambda_2, T) \cdot A = 8.916 \times 10^{-9} \text{ W}$$

Відповідь отримано в одиницях системи СІ. Система сама провела масштабування значень, які були визначені в різних одиницях, до однієї (мкм, см, м). Підкреслена зеленим змінна А показує, що відбулося перевизначення вже існуючих даних. В даному випадку – одиниці вимірювання – «Ампер».

*Приклад 3.5.5.* Розрахувати кількість квартир та будинків в мікрорайоні з населенням 1368 людей. В квартирі мешкає 2 особи, в будинку 60 квартир.

### РОЗВ'ЯЗАННЯ

1. Визначити базову позасистемну розмірність «людина» та похідні від неї розмірності «квартира», «дім»:

$$\text{людина} := 1 \quad \text{квартира} := 2\text{-лк дім} := 60\text{-квартира}$$

2. Ввести змінну x з кількістю людей. В маркер вписати базову розмірність, Вивести значення змінної. На місці маркера вписати розмірності «квартира», «дім»:

$$x := 1320\text{-людина} \quad x = 660\text{квартира} \quad x = 11\text{дім}$$

*Завдання 3.5.1.* Обрахувати щільність, питому вагу за умовами прикладу 3.5.2 в одиницях «г», «см», «сек».

*Завдання 3.5.2.* Обрахувати момент інерції

а) стрижня масою  $M=5 \text{ kg}$ , довжиною  $L=2 \text{ m}$ , який обертається відносно осі, що проходить через його кінець:

$$I = \frac{M \cdot L^2}{3}$$

б) прямокутної пластини масою  $M = 5 \text{ kg}$ , довжиною  $L = 2 \text{ m}$ , шириною  $h=10 \text{ cm}$ , відносно вісі, яка проходить через центр пластини перпендикулярно до неї:



$$I = \frac{M(h^2 + L^2)}{12}$$

*Завдання 3.5.3* Розрахувати кількість теплоти  $Q$  в кілоджоулях, яке виділиться при нагріванні праски масою  $M=1\text{кг}$  на  $dT=200^\circ\text{C}$  від  $20^\circ\text{C}$  до  $220^\circ\text{C}$ , якщо питома теплоємність матеріалу праски  $C=0.54 \text{ кДж/кг}\cdot\text{К}$  за виразом  $Q=C\cdot(T_2-T_1)\cdot M=C\cdot dT\cdot M$ .

## Практикум 3.6. Символьні перетворення

### 3.6.1. Теоретичні положення

*Символьними або аналітичними* обчисленнями називають обчислення, результат яких отримується не у вигляді чисел, а у вигляді формульних залежностей.

Вигляд результату залежить від конкретного завдання. Одна операція може розглядатися в одному випадку як вірна, а в іншому – ні. Крім того, комп'ютерні алгоритми недосконалі, тому відповіді системи в символьному вигляді є не завжди практичні, а іноді результат є простим повторенням вихідних даних.

Символьний процесор в *Mathcad* є запозиченим. В версіях системи до 12-ї застосовувався скорочений варіант символьного процесора з системи *Maple*, далі він був замінений на символьний процесор системи *MuPad*. Відповідно, можливості аналітичних перетворень в останніх версіях системи дещо скромніші.

В документах можливо аналітично проводити диференціювання, інтегрування, розкладання функції в ряд, спектральні перетворення, спрощування виразів, знаходження границі тощо.

Символьні дії можна провести «швидким» способом з меню **Symbolics**, способом «реального часу» кнопками панелі **Symbolic** (рис. 3.72) або оператором символьного виведення результату « $\rightarrow$ » (клавіші *Ctrl+.*).

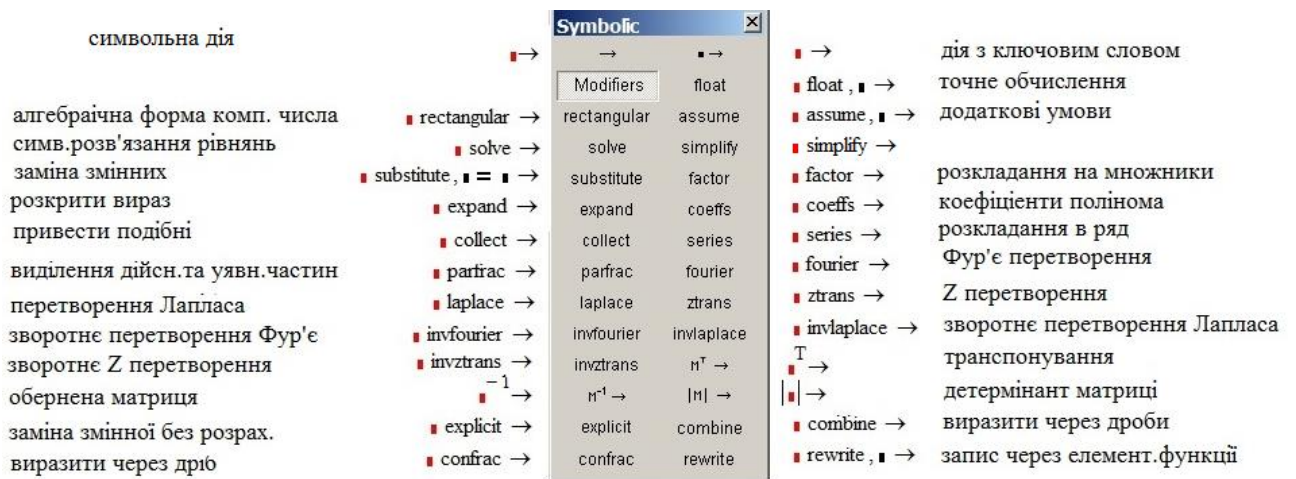


Рис. 3.72. Панель символічних перетворень

Запис перетворень «реального часу» складається з самого виразу, позначки символічного виведення та уточнюючого ключового слова. Ввести такий шаблон можна для кожної дії з панелі **Symbolic** та універсальною позначкою (рис. 3.73), яка викликається клавішами *Ctrl+Shift+*. Для деяких операцій після ключового потрібно вставляти для уточнення імена змін та вирази, у відношенні до яких планується проведення перетворень.



Рис. 3.73. Шаблон ключових слів

«Швидкий» спосіб є зручнішим, коли потрібно швидко отримати одноразово аналітичний результат без збереження ходу обчислень. Перетворення з меню відносяться тільки до одного виділеного виразу.

Спосіб «реального часу» дозволяє записувати вираз в традиційній формі з урахуванням виразів, що вже записані в документі та зберігати хід перетворень.

Розташування відповіді при «швидкому» способі (поруч, знизу, на місці завдання, наявність коментарів, що пояснюють здійснені операції) задається форматом перетворень в пункті **Evaluation style** меню **Symbolics**.

Для більшості символічних функцій дії мають наступний порядок:

1. Введення виразу.
2. Виділення змінної або виразу.

### 3. Виклик відповідної команди.

#### **Команди меню символної математики Symbolics**

Меню має 5 груп полів: група одиночних дій; група дій, що потребують виділення аргументів; група спектральних перетворень; група операцій з матрицями та службова група завдання формату вигляду результату.

#### **Група одиночних дій**

Пункт **Evaluate** дозволяє проводити одиночні обчислення в символному вигляді. Перед застосуванням пункту слід позначити вираз.

**Symbolically** – отримання результату в символному вигляді.

**Complex** – отримання результату в комплексному вигляді.

**Floating point** – отримання чисельного результату з великою кількістю знаків мантиси.

Пункт **Simplify** спрощує вираз.

Пункт **Expand** розкриває (розгортає) вираз.

Пункт **Factor** розкладає вираз, що виділено, на множники.

Пункт **Collect** є зворотнім до розкладання на множники.

Пункт **Polynomial Coefficients** обраховує коефіцієнти полінома, на який може бути розкладено вираз.

Група дій з виділеними змінними **Variable:**

**Solve** розв'язує рівняння.

**Substitute** проводить заміну змінних.

**Differentiate** диференціює вираз.

**Integrate** інтегрує вираз.

**Expand to Series** розкладає вираз в ряд Тейлора.

**Convert to Partial Fraction** розкладає вираз на частини.

Група матричних операцій **Matrix** дозволяє провести інвертування, транспонування матриці та обчислити її детермінант в символьному вигляді.

Група спектральних перетворень **Transform** виконує символьні перетворення Фур'є, Лапласа та дискретне z-перетворення.

### Символьне обчислення виразів

Отримання результатів обчислень може проводитися в чисельному або символьному вигляді. Символьні обчислення можуть проводитися для знаходження похідних, невизначених інтегралів, лімітів, сум та доданків, знаходження параметрів матриць.

Наприклад, для функції користувача  $f(x) := x^2 \cos(2x)$  застосування символу чисельного виведення дає чисельний результат  $f(3) = 8.642$ , застосування символу символьного виведення дає аналітичний результат  $f(3) \rightarrow 9 \cdot \cos(6)$ .

Для отримання результату обчислення в символьному вигляді слід:

1. Ввести вираз  $A \cdot \sin(\sin(B \cdot C))$ .
2. Зліва від виразу ввести символ правої стрілки з панелі **Symbolic**, пункту меню **Symbolics-Evaluate-Symbolically** або клавішами **Ctrl+**. На екрані з'явиться результат  $A \cdot \sin(\sin(B \cdot C)) \rightarrow A \cdot B \cdot C$

або

1. Ввести шаблон виразу для символьних обчислень символ правої стрілки з панелі або клавішами **Ctrl+**.  $\rightarrow$ .
2. На місце маркеру ввести вираз  $A \cdot \sin(\sin(B \cdot C)) \rightarrow A \cdot B \cdot C$ . На екрані з'явиться результат  $A \cdot \sin(\sin(B \cdot C)) \rightarrow A \cdot B \cdot C$ .

### Обчислення сум, добутоків, границі

Подібно до інтегрування/диференціювання проводиться обчислення нескінчених сум, добутків, лімітів.

Наприклад, для обчислення суми  $\sum_{n=0}^{\infty} \frac{1}{2^n}$ , добутку  $\prod_{n=2}^{\infty} 1 - \frac{1}{n^2}$ , ліміту  $\lim_{x \rightarrow 0} \left( \frac{\sin(ax)}{x} \right)$  слід:

1. З панелі **Calculus** ввести шаблони розширених операторів

$$\sum_{n=1}^{\cdot} \cdot \quad \prod_{n=1}^{\cdot} \cdot \quad \lim_{\cdot \rightarrow \cdot} \cdot$$

2. На місце маркерів ввести вихідні вирази та завершити введення символом «→». На екрані з'явиться результат

$$\sum_{n=0}^{\infty} \frac{1}{2^n} \rightarrow 2 \quad \prod_{n=2}^{\infty} \left( 1 - \frac{1}{n^2} \right) \rightarrow \frac{1}{2} \quad \lim_{x \rightarrow 0} \frac{\sin(a \cdot x)}{x} \rightarrow a$$

Примітка. Результат символічних перетворень залежить від версії системи та якості підготування вихідних даних. Наприклад, обчислення суми в версії 11 системи дає вірний результат:

$$\prod_{n=2}^{\infty} \left( \frac{n^3 - 1}{n^3 + 1} \right) \rightarrow \frac{2}{3}$$

, а вже в версії 14 результат важко назвати практичним:

$$\prod_{n=2}^{\infty} \left( \frac{n^3 - 1}{n^3 + 1} \right) \rightarrow \frac{2 \cdot \Gamma \left[ \frac{3}{2} - \left( \frac{\sqrt{3}}{2} \right) i \right] \cdot \Gamma \left( \frac{3}{2} + \frac{1}{2} \sqrt{3} i \right)}{\Gamma \left[ \frac{5}{2} - \left( \frac{\sqrt{3}}{2} \right) i \right] \cdot \Gamma \left( \frac{5}{2} + \frac{1}{2} \sqrt{3} i \right)}$$

За необхідності отримати чисельний результат з підвищеною точністю застосовується ключове слово **float**, праворуч від якого визначається необхідна кількість знаків мантиси результату. Наприклад, число  $\pi$  з 11-ма знаками система обраховує як

$$\pi \text{ float, 11} \rightarrow 3.1415926536$$

**Символьні тотожні перетворення виразів**

При математичних обчисленнях часто доводиться проводити перетворення виразів спрощуючи, групуючи їхні частини тощо. В системі такі операції потребують уточнення типу дії. Дії можна провести «швидким» способом з меню **Symbolics** та способом «реального часу» для збереження дії в документі.

### Спрощення виразів

Для спрощення виразів шляхом приведення до спільного знаменника, приведення подібних членів, тригонометричних перетворень застосовується ключове слово **simplify**.

Наприклад, для проведення спрощення виразу «швидким» способом слід:

1. Ввести вираз  $\sin(x)^2 + \cos(x)^2$ .
2. Обрати в меню **Symbolics** пункт **Simplify**.
3. В документі з'явиться результат  $\sin(x)^2 + \cos(x)^2$  simplifies to 1.

Для проведення спрощення виразу «в реальному часі» слід:

1. Ввести вираз  $\sin(x)^2 + \cos(x)^2$ .
  2. Обрати на панелі **Symbolic** кнопку **Simplify**.
  3. В документі з'явиться результат  $\sin(x)^2 + \cos(x)^2$  simplify → 1.
- Або
1. Вивести шаблон **Simplify** з панелі **Symbolic** **simplify** → .
  2. В маркер ввести вираз  $\sin(x)^2 + \cos(x)^2$ .
  3. В документі з'явиться результат  $\sin(x)^2 + \cos(x)^2$  simplify → 1.
- Або

1. Обрати на панелі **Symbolic** кнопку **Keyword**  .

2. В поле 1 ввести вираз  $\sin(x)^2 + \cos(x)^2$  , в поле 2 – ключове слово **Simplify**.

3. В документі з'явиться результат  $\sin(x)^2 + \cos(x)^2 \text{ simplify} \rightarrow 1$  .

Примітка. Система по різному виконує дії для аргументів з мантисою та без. Для аргументу без мантиси повертається сам вихідний вираз, для аргументу з мантисою проводиться чисельне обчислення:

$$\sqrt{3} \text{ simplify} \rightarrow \sqrt{3} \quad \sqrt{3.0} \rightarrow 1.7320508075688772935$$
$$\sqrt{3.0} \text{ simplify} \rightarrow 1.7320508075688772935$$

### Розкриття виразу

Для проведення розкриття виразу застосовується ключове слово **expand**. Дії спрощення та розкриття є зворотними. Наприклад,

$$2 \cdot \cos(x) \cdot \sin(x) \text{ simplify} \rightarrow \sin(2 \cdot x) \quad \sin(2 \cdot x) \text{ expand} \rightarrow 2 \cdot \cos(x) \cdot \sin(x)$$

$$(x + y)^3 \text{ expand} \rightarrow x^3 + 3 \cdot x^2 \cdot y + 3 \cdot x \cdot y^2 + y^3 \text{ simplify} \rightarrow (x + y)^3$$

В інструкцію **expand** можна вводити для уточнення вираз, який повинен залишитися в результаті без зміни. Наприклад, розкладання добутку

$(x - y) \cdot (x + y) \cdot (x^2 + y^2)$  дає правильний звичний результат  $(x - y) \cdot (x + y) \cdot (x^2 + y^2) \text{ expand} \rightarrow x^4 - y^4$  , розкладання за умови збереження множника  $x^2 + y^2$  слід записувати наступним чином

$$(x + y) \cdot (x - y) \cdot (x^2 + y^2) \text{ expand, } x^2 + y^2 \rightarrow x^2 \cdot (x^2 + y^2) - y^2 \cdot (x^2 + y^2)$$

### Розкладання на множники

Для дії розкладання на множники слугує ключове слово **factor**. Для виразів дія проводиться у вигляді множників-виразів. Таким чином можна знайти множники ЦІЛОГО числа, тобто визначити, чи є число простим.

Наприклад, щоб розкласти число 111 на множники можна ввести число, обрати дію **factor** та отримати результат  $111 \text{ factor} \rightarrow 3 \cdot 37$ .

Число 111 не є простим, бо складається з множників 1, 3, 37. А от число 113 є простим  $113 \text{ factor} \rightarrow 113$ .

Для виразів дія розкладання має наступний вигляд:

$$x^4 - 16 \text{ factor} \rightarrow (x - 2) \cdot (x + 2) \cdot (x^2 + 4)$$

Функція проводить розкладання на множники, якщо корені виразу є раціональними. У випадку ірраціональних коренів функція як результат повертає сам вираз. При необхідності отримати добуток, слід явно вказати ірраціональний корінь через кому після ключового слова **factor**.

Наприклад, для виразу  $x^2 - 2$  система не дасть добутку, бо коренем функції є  $x = \sqrt{2}$ :

$x^2 - 2 \text{ factor} \rightarrow x^2 - 2$ . Додавання значення кореню введе на екран очікуваний результат:

$$x^2 - 2 \text{ factor, } \sqrt{2} \rightarrow (x - \sqrt{2}) \cdot (x + \sqrt{2})$$

Примітка. Рекомендується для проведення перетворень використовувати кілька методів. Наприклад, для відношення  $\frac{1 - x^5}{1 - x}$  дія розгортання дає непрактичний результат без пониження ступеню

$$\frac{(1 - x^5)}{1 - x} \text{ expand} \rightarrow \frac{x^5}{x - 1} - \frac{1}{x - 1}$$

. Дія розкладання на множники не тільки розкладає різницю ступенів, а й скорочує ступінь

$$\frac{(1 - x^5)}{1 - x} \text{ factor} \rightarrow x^4 + x^3 + x^2 + x + 1$$

### Заміна змінної в виразі



Для аналізу або для розрахунків іноді доцільно отримати результат, коли в вихідному виразі заміненна змінна.

Нашвидкуруч для зміни змінної використовується пункт меню **Symbolics-Variable-Substitute**.

Наприклад, щоб отримати вираз  $\sin(kx^2+bx)$  із заміною  $k=ax^2$  слід:

1. Ввести вираз та вираз на заміну. Скопіювати вираз на заміну в кліпбоард (наприклад, клавішами *Ctrl+C*). В вихідному виразі виділити змінну **k**.

2. В меню обрати пункт **Symbolics-Variable-Substitute**. На екрані з'явиться результат:

$$\sin(k \cdot x^2 + b \cdot x) \text{ by substitution, yields } \sin(a \cdot x^4 + b \cdot x)$$

Для отримання такого ж результату способом «реального часу» слід:

1. Ввести вихідний вираз. На панелі обрати дію **Substitute**. В маркер ввести  $k=ax^2$ . На екрані з'явиться результат:

$$\sin(k \cdot x^2 + b \cdot x) \text{ substitute, } k = a \cdot x^2 \rightarrow \sin(a \cdot x^4 + b \cdot x)$$

Заміну змінних можна застосувати для перевірки отриманих відповідей. Наприклад, в якості розв'язання інтегрального рівняння  $\int_0^x \sin(t)dt = a$  отримано відповідь  $x=acos(a-1)$ . Для перевірки можна:

1. Записати відповідь  $x:=acos(a-1)$  та скопіювати її в кліпбоард.

2. Записати рівняння  $\int_0^x \sin(t)dt$ , виділити змінну **x**, викликати функцію **Substitute**.

3. На екрані з'явиться відповідь

$$\int_0^x \sin(t) dt \text{ by substitution, yields } a$$

Тобто  $x=acos(a-1)$  забезпечує значення інтеграла.

**Виведення алгебраїчної форми виразу**

Для отримання виразу, в якому явно виділені дійсна та уявна частини комплексу слугує функція **rectangular**. Наприклад,

$$\ln(1+i) \text{ rectangular} \rightarrow \frac{\ln(2)}{2} + i \cdot \left(\frac{\pi}{4}\right)$$

### Приведення подібних

Приведення подібних для елементарних функцій проводить дія **combine**. В якості елементарних функцій можна використовувати арктангенс, синус/косинус, логарифмічну функцію, гіперболічні синус/косинус. Їх визначають пояснюючі ключові слова: **atan**, **exp**, **ln**, **log**, **sincos**, **sinhcosh** відповідно. Пояснює слово додається через кому наприкінці. Наприклад,

$$\text{atan}(x) + \text{atan}(y) \text{ combine, atan} \rightarrow -\text{atan}\left[\frac{1}{(x \cdot y - 1)} \cdot (x + y)\right]$$

$$\sinh(x) \sinh(y) \text{ combine, sinhcosh} \rightarrow \frac{\cosh(x+y)}{2} - \frac{1}{2} \cdot \cosh(x-y)$$

### Розкладання виразу на алгебраїчний поліном

Функція **collect** намагається перегрупувати алгебраїчний вираз на алгебраїчний поліном з обраною змінною в якості аргументу так, щоб аргумент розташовувався з пониженням степеню. Введення імені змінної після ключового слова для функцій кількох змінних є обов'язковим.

Наприклад, застосування дії до виразу  $(x-4) \cdot (x-7) \cdot x + 99$  функції однієї змінної не потребує додаткового визначення змінної і дає наступну відповідь

$$(x-4) \cdot (x-7) \cdot x + 99 \text{ collect} \rightarrow x^3 - 11x^2 + 28x + 99$$

В результуючому поліномі аргумент  $x$  розташовується зі степенями з 3 до 0 послідовно.

Для функції кількох змінних звернення може бути наступним:

$$x \cdot y + x^2 + y^2 + x^2 \cdot y^2 \text{ collect, x} \rightarrow (y^2 + 1) \cdot x^2 + y \cdot x + y^2$$

$$x \cdot y + x^2 + y^2 + x^2 \cdot y^2 \text{ collect, y} \rightarrow (x^2 + 1) \cdot y^2 + x \cdot y + x^2$$

Результат дії **collect** не завжди очікувані. Деякі приклади наведені нижче. В першому прикладі функція дає результат, який не є поліномом. В другому прикладі в результаті не приведені подібні, хоча функція розкладання видає результат не в порядку зменшення степенів  $x$ , але з приведенням подібних.

$$\frac{1}{x} + \frac{1}{x-1} \text{ collect} \rightarrow -\frac{2-x-1}{x-x^2}$$

$$\left(x + \frac{y^2}{x}\right)^3 \text{ collect, x} \rightarrow \frac{x^6 + 3 \cdot x^4 \cdot y^2 + 3 \cdot x^2 \cdot y^4 + y^6}{x^3} \quad \left(x + \frac{y^2}{x}\right)^3 \text{ expand} \rightarrow 3 \cdot x \cdot y^2 + x^3 + \frac{3 \cdot y^4}{x} + \frac{y^6}{x^3}$$

### Розрахунок коефіцієнтів поліному

Для випадків, коли не потрібен весь алгебраїчний поліном, в який розкладається вираз, а тільки його коефіцієнти, слугує функція **coeffs**. Функція повертає вектор, в якому за збільшенням ступеню аргументу розташовані коефіцієнти від початкового ступеню аргументу на першому місці до найбільшого – на останньому. Викликається функція **coeffs** аналогічно інструкції **collect**.

Наприклад, добуток з прикладу, що наведено вище відбивається на поліном  $x^3 - 11x^2 + 28x + 99$ . Тобто, має коефіцієнти 1 -11 28 99. Виклик функції дає наступний результат

$$(x-4) \cdot (x-7) \cdot x + 99 \text{ coeffs} \rightarrow \begin{pmatrix} 99 \\ 28 \\ -11 \\ 1 \end{pmatrix}$$

Всього вектор містить чотири елементи, тобто поліном має третій ступінь. Перший елемент вектора 99 є коефіцієнтом для  $x^0$ , другий 28 – для  $x$ , третій -11 для  $x^2$ , останній 1 для  $x^3$ .

### Розкладання в ряд Тейлора

Для отримання розкладання функції в ряд Тейлора призначена функція **series**. Викликається функція **series** аналогічно інструкції **collect**. Після функції слід визначати через кому ім'я змінної по якій проводиться розкладання в ряд. За замовчанням розкладання проводиться в

околі точки 0. Після ключового слова можна визначити максимальний ступінь доданків ряду або значення аргументу, навколо якого буде проведено розкладання.

Наприклад, для розкладання в ряд Тейлора експоненційної функції за замовчанням в околі точки 0 слід наступним чином застосувати функцію **series**

$$e^x \text{ series} \rightarrow 1 + x + \frac{x^2}{2} + \frac{x^3}{6} + \frac{x^4}{24} + \frac{x^5}{120} ,$$

для отримання 3-х доданків виклик має виглядати наступним чином

$$e^x \text{ series}, 3 \rightarrow 1 + x + \frac{x^2}{2} ,$$

для того, щоб розкладання пройшло в околі точки  $x=2$ :

$$\exp(x) \text{ series}, x = 2 \rightarrow e^2 + e^2 \cdot (x - 2) + \frac{e^2 \cdot (x - 2)^2}{2} + \frac{e^2 \cdot (x - 2)^3}{6} + \frac{e^2 \cdot (x - 2)^4}{24} + \frac{e^2 \cdot (x - 2)^5}{120} .$$

### Ланцюги обчислень

Система далеко не завжди видає очікуваний результат. Часто доводиться його повторно спрощувати чи перетворювати. Для цього можна вручну скопіювати отриманий вираз та застосувати потрібну команду, але можна й запустити кілька символічних дій послідовно. Додавання додаткових дій проводиться вставлянням рядка дії кнопкою панелі після основної дії.

Наприклад, потрібно розкласти експоненційну функцію в ряд Тейлора з трьома доданками та обчислити її значення при  $x=2$ .

Можливо послідовне застосування дій розкладання в ряд Тейлора з кількістю доданків 3 функцією **series** та заміна змінної функцією **substitute**:

1. Ввести експоненційний вираз, ввести шаблон функції **series**, визначити кількість доданків 3.

2. Поки блакитний курсор – кут знаходиться на правому кінці рядка **series**

$e^x \text{ series}, 3 \rightarrow$

клацнути по позначці **Substitute** на панелі **Symbolic**

$$e^x \left| \begin{array}{l} \text{series, 3} \\ \text{substitute, } y = \end{array} \right. \rightarrow$$

3. Заповнити маркер рядка **Substitute**. На екрані з'явиться результат

$$e^x \left| \begin{array}{l} \text{series, 3} \\ \text{substitute, } x = 2 \end{array} \right. \rightarrow 5$$

### Запис виразу через елементарні функції

Отримання виразу, в якому вихідний вираз представлений через елементарні функції: *exp*, *ln*, *log*, *sincos* (*sin*, *cos*), *sinhcosh* (*sinh*, *cosh*), – забезпечує функція **rewrite**. Функція, через яку буде проводитися опис, визначається ключовим словом через кому після слова **rewrite**. Наприклад,

$$\operatorname{asinh}(x) \operatorname{rewrite}, \ln \rightarrow \ln[x + \sqrt{x^2 + 1}].$$

### Розкладання виразу на суму правильних дробів

Для розкладання поліноміального виразу на суму правильних дробів слугує функція **parfrac**. Кількість доданків можна визначити після коми числом. Якщо в виразі присутні кілька змінних, то обов'язковим є уточнення імені змінної, для якої буде проводитися дія. За замовчанням ряд містить раціональні корені виразу. За необхідності врахування ірраціонального значення, слід додатково вписати його через кому.

Наприклад,

$$\frac{1}{(x^2 + 1)(x^2 - 1)} \operatorname{parfrac} \rightarrow \frac{1}{4 \cdot (x - 1)} - \frac{1}{2 \cdot x^2 + 2} - \frac{1}{4 \cdot x + 4}$$

$$\frac{3a}{(x^2 - 8x + 15)} \operatorname{parfrac}, x \rightarrow \frac{3 \cdot a}{2 \cdot (x - 5)} - \frac{3 \cdot a}{2 \cdot x - 6}$$

$$\frac{1}{3(x^2 - 2)} \operatorname{parfrac}, \sqrt{2} \rightarrow \frac{\sqrt{2}}{12 \cdot (x - \sqrt{2})} - \frac{\sqrt{2}}{12 \cdot x + 12 \cdot \sqrt{2}}$$

### Вимкнення змінної з розрахунку

При використанні способу «реального часу» коли змінним попередньо задані якісь значення, система обраховує вирази з урахуванням цих чисельних значень. Тобто, результат може виявитись чисельним. Для того, щоб виключити кількісне значення змінних з символічних перетворень застосовується функція **explicit**. Імена змінних, які вважаються символічними, перераховуються через кому після ключового слова.

Наприклад, якщо змінні попередньо визначені як  $a:= 3.7$   $b:= 12.1$   $c:= \sin(1.2\pi)$ , то символічне обрахування дасть в результаті обрахунку число

$$2 \cdot a + b - c \rightarrow 20.0877852522924731292$$

Для того, щоб отримати вираз без чисельних значень змінної «с» слід ввести:

$$2 \cdot a + b - c \text{ explicit } , a, b \rightarrow 2 \cdot 3.7 + 12.1 - c$$

Наприклад, якщо змінній «х» завдано попереднє значення  $x:= 5$ , то розкладання на множники дасть в результаті обрахунку число

$x^2 - 3x + 2$  **factor**  $\rightarrow 2 \cdot 3$ . Для того, щоб отримати вираз без значень змінних слід ввести:

$$x^2 - 3x + 2 \left| \begin{array}{l} \text{explicit} \\ \text{factor} \end{array} \right. \rightarrow (x - 1) \cdot (x - 2)$$

### Обчислення виразу у вигляді правильного нескінченного дроби

Для запису виразу у вигляді нескінченного дроби (рис. 3.74) використовується функція **confrac**.

$$a_1 + \frac{1}{a_2 + \frac{1}{a_3 + \dots}}$$

$$\dots + \frac{1}{a_{n-1} + \frac{1}{a_n}}$$

Рис. 3.74 Нескінчений дріб

Результатом є вектор  $[a_1 \ a_2 \ a_3 \ \dots \ a_n]$ . За замовчанням враховується 10 кроків ділення. Змінити кількість можна визначенням цілого числа після коми. Для того, щоб вивести не вектор, а сам дріб, слід додатково ввести ключове слово **fraction**.

Наприклад,

$$\frac{14 + \sqrt{5}}{3} \text{ confrac,3} \rightarrow \begin{pmatrix} 5 \\ 2 \\ 2 \\ 2 \end{pmatrix} \frac{14 + \sqrt{5}}{3} \text{ confrac,3,fraction} \rightarrow 5 + \left[ 2 + \frac{1}{\left( 2 + \frac{1}{2} \right)} \right]$$

### Обчислення за додаткових умов

Система допускає виконання послідовності символічних дій та встановлення додаткових умов при виконанні перетворень.

Для визначення обмежень при перетвореннях застосовується ключове слово **assume**. Слово вставляється в наступний після рядка основної дії рядок кнопкою з панелі **Symbolic**.

З **assume** застосовуються наступні логічні вирази: умови більшості або меншості  $>$ ,  $<$ ,  $\geq$ ,  $\leq$ , рівності значенню, рівності типу **real**, **integer**, **complex**, рівності типу в діапазоні  $a < x < b$  **RealRange(a,b)**. Типи обираються з панелі **Modifier** або вводяться вручну. Декілька умов записуються через кому. Якщо умова стосується всіх змінних виразу, застосовується службове слово **ALL**. Наприклад,

$$\sqrt{x^2} \left| \begin{array}{l} \text{simplify} \\ \text{assume, } x = \text{real} \end{array} \right. \rightarrow |x| \quad \sqrt{x^2} + \sqrt{y^2} + \sqrt{z^2} \left| \begin{array}{l} \text{simplify} \\ \text{assume, ALL} > 0 \end{array} \right. \rightarrow x + y + z$$

Символьні дії в *MathCAD Prime* реалізовані способом «реального часу».

Запис перетворень «реального часу» складається з вихідного математичного виразу, позначки символічного виведення та уточнюючого ключового слова. Деякі операції після ключового слова потребують уточнення імені змінної та режиму проведення перетворень. Введення даних для символічних перетворень вводиться через панель **Math-Symbolic** (рис. 3.75), яка містить всі вбудовані елементи керування. Особливістю складу панелі є те, що більшість ключових слів, які уточнюють режим проведення перетворень не вводяться вручну, а беруться з розділу модифікаторів панелі **Modifiers**.

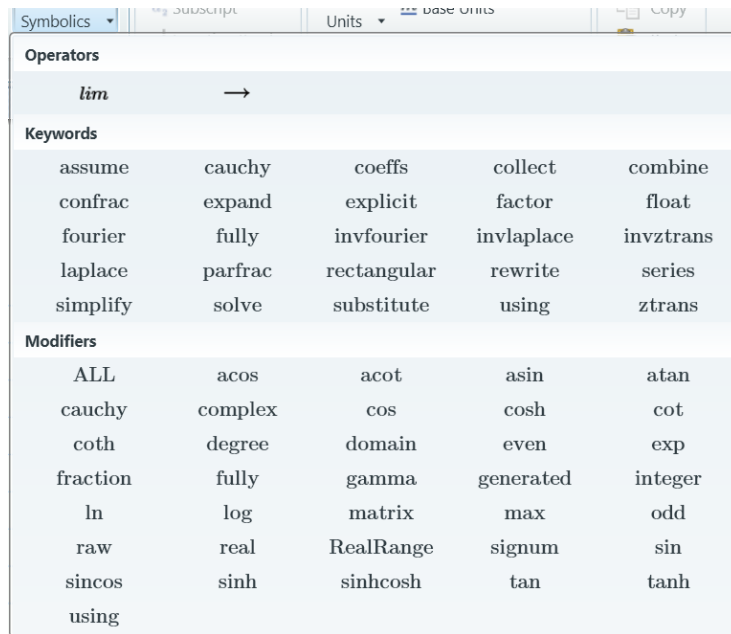


Рис. 3.75. Панель символічних перетворень

Натискання кнопки панелі символічних дій виводить на екран шаблон відповідної символічної дії.

Універсальний шаблон символічних перетворень вводиться символом правої стрілки «→» (*Ctrl+.*) (рис. 3.76).

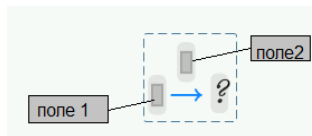


Рис. 3.76. Універсальний шаблон

В поле 1 вводиться математичний вираз або ім'я функції користувача, в поле 2 вводиться з клавіатури ключове слово та, за потреби уточнення.

Наприклад, провести спрощення  $\sin(x)^2 + \cos(x)^2$  виразу можна кількома способами.

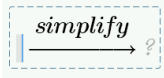
Спосіб 1.

- Ввести вираз  $\sin(x)^2 + \cos(x)^2$  та символ «→».
- В поле 2 ввести слово **simplify**. На екрані з'явиться результат



-  $\cos(x)^2 + \sin(x)^2 \xrightarrow{\text{simplify}} 1$

Спосіб 2.

- Ввести шаблон дії спрощення з панелі стрічки  .
- В поле 1 ввести вираз  $\sin(x)^2 + \cos(x)^2$  .

### 3.6.1. Завдання для самостійного виконання

Завдання 3.6.1. Спростити вираз

а)  $\frac{3x^2 - 2x}{6 - 7x - 3x^2}$  б)  $\frac{a - \frac{4a - 4}{a}}{\frac{2}{a} - 1}$  в)  $\frac{x - 3}{4x^2 + 24x + 36} \cdot \left( \frac{x}{3x - 9} - \frac{3}{9 + 3x} + \frac{x^2 + 9}{27 - 3x^2} \right)$  .

Завдання 3.6.2. Розкласти вираз на множники

а)  $x^4 - 10x^3 + 35x^2 - 50x + 24$  б)  $a^2b + b^2a + 2abc + b^2c + a^2c + ac^2 + bc^2$  .

Завдання 3.6.3. Розгорнути вирази

а)  $\cos(5x)$  б)  $(2x^3 + 3x^2 + 4x + 1)(x - 1)(x - 2)(x - 3)$  .

Завдання 3.6.4. Розкласти в ряд Тейлора функцію до 5-го ступеню

а)  $\sin^2(x)$  б)  $\text{tg}(x)$  .

Завдання 3.6.5. Визначити, чи є числа 123456789, 2127 -1 простими.

Завдання 3.6.6. Визначити збіжність ряду. Ряд сходиться, якщо  $|a_n| < 1$  за  $n \rightarrow \infty$ . В разі збіжності, розрахувати суму ряду.

а)  $\sum_{n=1}^{\infty} \frac{2^{n+1}(3n+2)}{n!}$  б)  $\sum_{n=1}^{\infty} \frac{n!}{n^2 2^n}$  в)  $\sum_{n=1}^{\infty} \frac{n^3}{e^n}$  г)  $\sum_{n=1}^{\infty} \frac{n}{(2n+1)!}$  .

Завдання 3.6.7. Розкласти функцію  $e^x$  в ряд Тейлора до 3-го ступеню та обчислити значення ряду для  $x=1$  .

## Практикум 3.7. Обмін даними

### 3.7.1. Теоретичні положення

В продуктах *MathCAD* проводиться засобами файлового обміну та об'єктними компонентами. Імпорт-експорт може бути проведений програмуванням дій з даними з використанням вбудованих функцій та застосуванням засобів технології зв'язування та передавання об'єктів (*Object Linking and Embedding*).

#### Компонент Excel

Компонент *Excel* – це спосіб поєднання застосунків *Excel* та *MathCAD*. В документі *MathCAD* компонент відображається як таблиця *Excel*. Вставлення компонента *Excel* супроводжується одночасним відкриттям застосунку *MS Excel*, тому відсутність на комп'ютері інстальованого унеможливорює роботу з компонентом *Excel* в *MathCAD*.

Компонент *Excel* забезпечує одночасний доступ до таблиці з *MathCAD* та з *MS Excel*. Можна редагувати та формувати дані за допомогою всіх інструментів *MS Excel* в документі *MathCAD*. Компонент *Excel* створює один аркуш із назвою *Sheet1*. Для посилання на інші аркуші потрібно створити аркуші в *MS Excel*.

Компоненти *Excel* можуть використовувати можливості обчислення *MS Excel* у *Mathcad*. Розрахунок *MS Excel* може бути частиною потоку обчислення *Mathcad*. У вбудований аркуш *Excel* можна надіслати значення та витягти з нього значення.

Для роботи з компонентом *Excel* потрібен *MS Excel 2007* або пізнішої версії.

В самому *MS Excel* можна зберегти дані з компонента *Excel* як зовнішній файл *.xlsx*. Однак між збереженим файлом та компонентом *Excel* зв'язок не зберігається.

Кожен компонент *Excel* містить три області: *входи (Inputs)*, *таблицю*, *виходи (Outputs)*. Вирази області *входів* визначають дані, які надсилаються з *MathCAD* до таблиці *MS Excel*. Вхідними даними можуть слугувати чисельні значення, змінні *Mathcad* – скаляри, вектори, матриці, рядки. Входу

можна призначити одну комірку або діапазон комірок, які будуть приймати дані *Mathcad*. Кількість *входів* не обмежується.

*Таблиця* компоненту є зображенням повної таблиці *MS Excel*. Подвійний щиглик на таблиці активує *MS Excel* та передає йому керування.

Вирази області *виходів* визначають дані, які надсилаються з *MS Excel* в *Mathcad* та присвоюються змінним *Mathcad*. Кількість виходів не обмежується.

Вирази *входів* та *виходів* за замовчуванням стосуються комірки A1 першого аркуша *MS Excel*.

Зміни в таблиці *MS Excel* негайно відображаються в таблиці компоненту *Excel Mathcad*. Однак обчислення вхідних та вихідних виразів НЕ ВІДБУВАЄТЬСЯ доки застосунок *MS Excel* відкритий. Оновлення даних відбувається тільки при закритті *MS Excel*. Входи та виходи компонента, який вставлений в колонити документа *Mathcad* НЕ обробляються.

### Програмний файловий обмін

Пакет має вбудовані функції для обміну даними з текстовими файлами формату *ASCII* з розділенням даних пробілами або знаком табуляції, з текстовими файлами формату *ASCII* з розділенням даних комою (файли *CSV*), з файлами *Excel 2007* та вище формату *Open XML* (файли *xlsx*). В якості даних для обміну слугують одномірні масиви вектори-стовпці та вектори-рядки або двомірні масиви – матриці.

### Програмні засоби обміну з текстовими файлами

#### Засоби MathCAD

Для роботи з текстовими файлами призначені функції **WRITEPRN**, **READPRN**, **APPENDPRN**, **READCSV**, **WRITECSV**, **WRITEFILE**, **READFILE**.

Функція **M:=READPRN(file)** читає в масив **M** з текстового *ASCII* файлу дані розділені пробілами або знаками табуляції.

Функція **var:=WRITEPRN(file,M)** записує масив **M** в *ASCII* файл з розділенням пробілами.

Функція **var:=APPENDPRN(file,M)** дописує масив **M** в *ASCII* файл з розділенням пробілами з розширенням вниз.

Обов'язковий аргумент **file** є текстовим рядком в лапках з повною специфікацією файлу. Наприклад, "d:\student\file\_output.txt".

Формат даних для функцій **WRITEPRN** та **APPENDPRN** визначається системними змінними **PRNPRECISION** та **PRNCOLWIDTH**.

Функція **WRITECSV(M, file)** записує масив **M** в текстовий ASCII файл з розділенням комою (файли **.CSV**). Функція є доступною для *MathCAD 15*.

Функція

**M:=READCSV(file, <rows>, <cols>, <emptyfill>, <delimiter>)** читає в масив **M** з текстового ASCII файлу дані з розділенням визначеним символом. Функція доступна для *MathCAD 15*.

Опціональний аргумент **rows** показує діапазон рядків для зчитування. Індеси рахуються з 1 незалежно від значення **ORIGIN**. Є числом або двоелементним вектором. Числом визначається кількість рядків, які будуть зчитані з початку файлу. В векторі діапазон визначається через номер першого зчитаного рядка, який задається в першому елементі вектора, та номер останнього зчитаного рядка, який задається в другому елементі вектора.

Опціональний аргумент **cols** показує діапазон стовпців для зчитування. Нумерація індесів та форма визначення повторює аргумент **rows**.

Опціональний аргумент **emptyfill** визначає символ, який буде замінювати в результуючому масиві **M** незаповнені комірки файлу. Може бути скаляром (числом), текстовим рядком або символом **NaN**. За замовчанням **NaN**.

Опціональний аргумент **delimiter** у вигляді односимвольного текстового рядка, який визначає символ-розділювач даних. Наприклад, ";". За замовчанням – кома.

Функція **WRITEFILE(M, file, type, <header>)** записує масив **M** в файл визначеного типу. Функція доступна для *MathCAD 14*.

Аргумент **type** є текстовим рядком з визначенням типу файлу. Можливі значення "**delimited**" – текстові дані з розділителем стовпців комою для файлів типу *CSV*, "**fixed**" – дані з стовпцями однакової ширини, "**Excel**" – файли MS Excel. "**Unicode**" – текстові дані з кодуванням *UTF-8*.

Опційний аргумент **header** є текстовим рядком. Може використовуватися для введення в файл інформації з коментарем, описом даних тощо.

Функція

```
M:=READFILE(file, type, <colwidths>, <rows>, ...  
<cols>, <emptyfill>)
```

зчитує в масив **M** дані з файлу визначеного типу.

Опціональний аргумент **colwidths** застосовується для файлів типу **fixed**. Є вектором-рядком розміром  $n \times 1$ , де  $n$  – загальна кількість стовпців в файлі. Визначає ширину стовпця в символах.

Функція **APPENDFILE**(**M**, "file", "type") додає значення матриці **M** до існуючого файлу **file**. Функція доступна для *MathCAD 14*.

### Засоби MathCAD Prime

Для роботи з текстовими файлами призначені функції **WRITEPRN**, **READPRN**, **APPENDPRN**, **READCSV**, **WRITECSV**, **READFILE**, **WRITEFILE**, **WRITETEXT**, **READTEXT**.

Функція **M:=READPRN**(file) читає в масив **M** з текстового ASCII файлу дані розділені пробілами або знаками табуляції.

Функція **var:=WRITEPRN**(file, **M** <, row, col>) записує масив **M** в ASCII файл з розділенням пробілами.

Функція **var:=APPENDPRN**(file, **M**) дописує масив **M** в ASCII файл з розділенням пробілами з розширенням вниз.

Обов'язковий аргумент **file** є текстовим рядком в лапках з повною специфікацією файлу. Наприклад, "**d:\student\file\_output.txt**".

Опціональний аргумент **rows** показує діапазон рядків для запису. Індокси рахуються з 1 незалежно від значення **ORIGIN**. Є числом або

двоелементним вектором. Числом визначається початковий рядок, з якого будуть зчитані дані з матриці. Зчитування проходить до кінця матриці. В векторі діапазон визначається через номер першого зчитаного рядка, який задається в першому елементі вектора, та номер останнього зчитаного рядка, який задається в другому елементі вектора.

Опціональний аргумент **cols** показує діапазон стовпців для зчитування. Нумерація індексів та форма визначення є аналогічною аргументу **rows**.

Функція **WRITECSV(file, M, <row, col, dec>)** записує масив **M** в текстовий ASCII файл з розділенням комою.

Опціональний аргумент **dec** визначає символ десяткової крапки в числах. Можливими значеннями є кома "," або крапка "." (за замовчанням). Кома робить символом-розділювачем символ крапки з комою ";".

Функція **M:=READCSV(file)** читає в масив **M** з текстового ASCII файлу дані, які в рядках розділені комою.

Функція **WRITEFILE(file, M)** записує масив **M** в файл з розділенням табуляцією.

Функція

```
M:=READFILE(file, type, <colwidths>, <rows>, ...  
<cols>, <emptyfill>, <dec>)
```

зчитує в масив **M** дані з текстового файлу визначеного типу.

Опціональний аргумент **emptyfill** визначає символ, який буде замінювати в результуючому масиві **M** незаповнені комірки файлу. Може бути скаляром (числом), текстовим рядком або символом **NaN**. За замовчанням **NaN**.

Аргумент **type** є текстовим рядком з визначенням типу файлу. Можливі значення "**delimited**" для текстових даних з розділителем стовпців комою або пробілом, "**fixed**" для даних із стовпцями однакової ширини.

Опціональний аргумент **colwidths** застосовується для файлів типу **fixed**. Є вектором-рядком розміром  $n \times 1$ , де  $n$  – загальна кількість стовпців в файлі. Визначає ширину стовпця в символах.

Опціональний аргумент **dec** заявлено в документації ([https://support.ptc.com/help/mathcad/en/index.html#page/PTC\\_Mathcad\\_Help/data\\_files\\_2.html](https://support.ptc.com/help/mathcad/en/index.html#page/PTC_Mathcad_Help/data_files_2.html)), але на практиці не є працездатним.

Функція

**A=READTEXT(file, <type, delimiter, colwidths, row, col, emptyfill, blank, options>)** зчитує дані в масив A з ASCII файлу.

Функція **WRITETEXT(file, M, <rows, cols, dec>)**—записує значення матриці M в текстовий файл .

Опціональний параметр **delimiter** визначає символ-розділювач. Наприклад, " , " , " " , "\t" (табуляція) тощо.

Опціональний параметр **row/col** є текстовим рядком, який визначає діапазон зчитування. Один рядок/стовпець задається числом "2", діапазон задається початком та кінцем через тире як "2-6", діапазон від рядка/стовпця до кінця задається як «2-". Значення можуть задаватися через кому.

Опціональний параметр **blank** є текстовим рядком, який визначає дії з незаповненими рядками. Можливими значеннями є пропустити "**skip**", зчитати "**read**" (за замовчанням), зупинитися "**stop**".

Опціональний параметр **options** є триелементним вектором-стовпцем з елементами текстовими рядками. Перший елемент визначає текст. Припустимими значеннями є " , ' або **none**. Другий елемент визначає десятковий розділювач. Можливими значеннями є кома або точка (за замовчанням). Третій елемент визначає розділювач порядків числа. Можливими значеннями є точка , кома, пробіл.

## Програмні засоби обміну з MS Excel

### Засоби MathCAD

Функція **A=READEXCEL(file, range <, emptyfill>)** зчитує в масив А дані з файлу *MS Excel* формату **.xlsx**. Функція доступна для *MathCAD 15*.

Оператор **WRITEEXCEL(M, file, range)** записує дані з масиву М в файл *MS Excel*. Функція доступна для *MathCAD 15*.

Аргумент **range** є текстовим рядком, який визначає діапазон даних. Записується у форматі *MS Excel*. Наприклад, "**Sheet1!A1:B3**".

Функції обміну **WRITEEXCEL**, **READEXCEL** *MathCAD* працюють тільки з тими типами даних, які обробляє *MS Excel*, вони не сприймають комплексні числа. Діапазон для запису слід задавати однією адресою, яка означає адресу початку вставляння даних в таблицю *MS Excel*. Значення **NaN** записуються в як порожня комірка, вирази перед записом обраховуються. Для запису можна використовувати тільки англійські імена аркушів книги за замовчанням на зразок **Sheet3**.

### Засоби MathCAD Prime

Функція **A=READEXCEL(file <, range, emptyfill)** зчитує в масив А дані з файлу *MS Excel* формату **.xlsx**.

Оператор **WRITEEXCEL(file, M <, rows, cols, range>)** записує дані з масиву М в файл *MS Excel*.

Опціональний аргумент **range** є текстовим рядком, який визначає діапазон даних. Може записуватися у форматі *MS Excel*. Наприклад, "**Sheet1!A1:B3**". Аркуш книги може визначатися його номером в квадратних дужках. Наприклад, "**[1]A1:B3**". Визначення однієї адреси для функції означає зчитування однієї вказаної комірки. Визначення однієї адреси для функції означає запис зчитування однієї вказаної комірки.

Опціональний аргумент **emptyfill** є текстовим рядком або числом, на яке будуть замінені нерозпізнані файлові дані. За замовчанням **NaN**.



Опціональний аргумент **rows** показує діапазон рядків для запису. Індеси рахуються з 1 незалежно від значення **ORIGIN**. Є числом або двоелементним вектором. Числом визначається кількість рядків, які будуть взяті з матриці. В векторі діапазон визначається через номер першого взятого рядка, який задається в першому елементі вектора, та номер останнього взятого рядка, який задається в другому елементі вектора.

Опціональний аргумент **cols** показує діапазон стовпців для зчитування. Нумерація індесів та форма визначення є аналогічною аргументу **rows**.

Функції обміну **WRITEEXCEL**, **READEXCEL** *MathCAD Prime* працюють з всіма типами чисел, з комплексними включно. Значення **NaN** записуються в як порожня комірка, вирази перед записом обраховуються. Для запису можна використовувати тільки англійські імена аркушів книги за замовчанням на зразок **Sheet3**.

## Інтерактивні засоби файлового обміну

### Засоби MathCAD

Інтерактивні засоби активуються з пунктів *Insert-Data*, *Insert-Component* меню (рис. 3.76). Всі дії проводяться у віконному діалозі.

Користувач може обрати засоби файлового імпорту *File Input*, засоби файлового експорту *File Output*, засоби універсального імпорту у вигляді універсальних компонентів *Data Wizard*, компонент *MS Excel*.

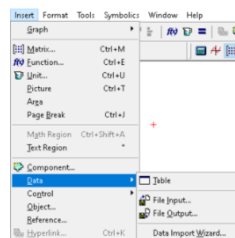


Рис. 3.76. Інтерактивні засоби файлового обміну

Слід зазначити, що вказані засоби зберігають зв'язок між документом *MathCAD* та файлом. Щигликом на позначці в документі можна оновити інформацію з файлу.

Пункт контекстного меню компоненту відкриває вікно властивостей компоненту. Сторінки вікна повторюють вікна діалогу.

### Імпорт з текстових файлів

Режим імпорту даних із зовнішнього файлу запускається пунктом **Insert-Data-File Input** меню. В першому вікні (рис. 3.77 а) пропонується визначити тип файлу. Можливий вибір текстових файлів *Text*, файлів *MS Excel*, файлів баз даних *dBase III*, файлів *Matlab*. В цьому ж вікні можна визначити вигляд десяткового розділителя у вигляді коми *use comma as decimal separator* та обрати файл. В наступному вікні (рис. 3.77 б) визначається діапазон даних в файлі для зчитування. По завершенню діалогу в документі *MathCAD* з'являється позначка дискової операції з одним маркером (рис. 3.77в). В цей маркер потрібно ввести ім'я матриці, в яку будуть імпортовані дані. На позначці в документі в подальшому щигликом миші можна оновити інформацію з файлу.

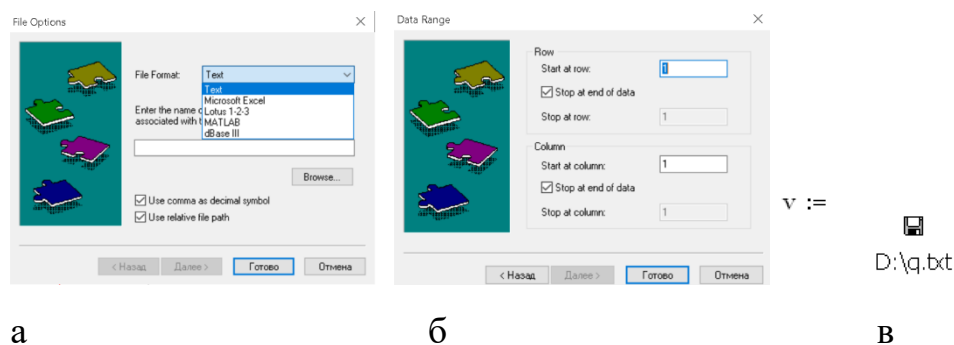


Рис. 3.77. Імпорт файлової: а – вікно визначення файлу; б – вікно визначення діапазону; в – позначка в документі

### Імпорт файлів MS Excel

Імпорт файлів *MS Excel* проводиться аналогічно імпорту з текстових файлів. В першому вікні (рис. 3.77 а) обирається тип файлу *MS Excel*. В наступному вікні (рис. 3.78 а) визначається діапазон даних в файлі для зчитування та аркуш книги *MS Excel*. Аркуш книги обирається із списку **Worksheet** згори вікна. Діапазон комірок визначається у форматі *MS Excel*

на зразок **Sheet1!A1:C4**. По завершенні діалогу в документі *MathCAD* з'являється позначка дискової операції з одним маркером (рис. 3.78 б). В цей маркер потрібно ввести ім'я матриці, в яку будуть імпортовані дані. На позначці в документі в подальшому щигликом миші можна оновити інформацію з файлу.

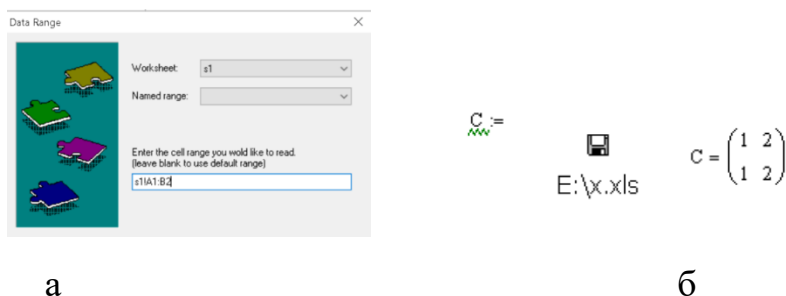


Рис. 3.78. Імпорт з *MS Excel*: а – вікно визначення діапазону; б – позначка в документі

### Файловий експорт

Режим експорту даних в зовнішній текстовий або *MS Excel* файл запускається пунктом **Insert-Data-File Output** меню. В першому вікні (рис. 3.79 а) пропонується визначити тип файлу. Додатково до опцій імпорту можна вибрати форматований текст *Formatted Text*, текст з розділенням табуляцією *Tab Delimited Text* та текст з розділенням комою *Comma Separated Text*. В наступному вікні (рис. 4.79 б) визначається діапазон даних в файлі для зчитування. По завершенню діалогу в документі *MathCAD* з'являється позначка дискової операції з одним маркером (рис. 3.79 в). В цей маркер потрібно ввести ім'я матриці, з якої будуть експортовані дані.

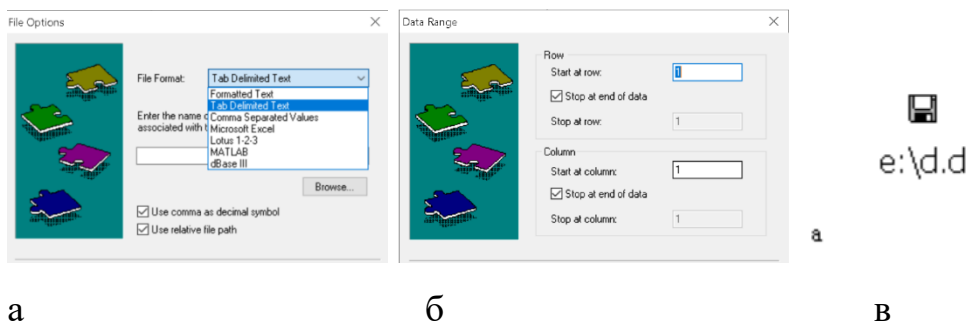


Рис. 3.79. Експорт файловий: а – вікно визначення файлу; б – вікно визначення діапазону; в – позначка в документі

Слід зазначити, що вказаний засіб експорту не дозволяє обрати аркуш книги та адреси на аркуші *MS Excel* для запису.

## Універсальний імпорт

Засіб універсального імпорту створює в документі не матрицю, а особливий об'єкт – компонент. Відображається компонент у вигляді таблиці. Доступ до елементів компоненту та дії з елементами компоненту нічим не відрізняється від доступу та роботи з елементами матриць.

*Data Import Wizard* із зовнішнього файлу запускається пунктом **Insert-Data- Data Import Wizard** або **Insert-Component-Data Import Wizard** меню.

В першому вікні (рис. 3.80 а) пропонується визначити тип файлу. Можливий вибір текстових файлів з розділювачем *Delimited Text*, текстових файлів зі стовпцями фіксованої ширини *Fixed Width Text*, файлів *MS Excel*, файлів баз даних *dBase III*, файлів *Matlab*. бінарних файлів *Binary*. В цьому ж вікні можна обрати файл.

Подальші дії визначаються обраним типом файлу.

### Імпорт тексту з розділенням

Друге вікно **Delimited Text Options** (рис. 3.80 б) містить відображення файлових даних, поля для визначення символу розділення *Delimiter*, початкового рядку *Starting row*, дій з порожніми рядками *Blank rows*, кількості рядків колонтитулу *Footer rows*.

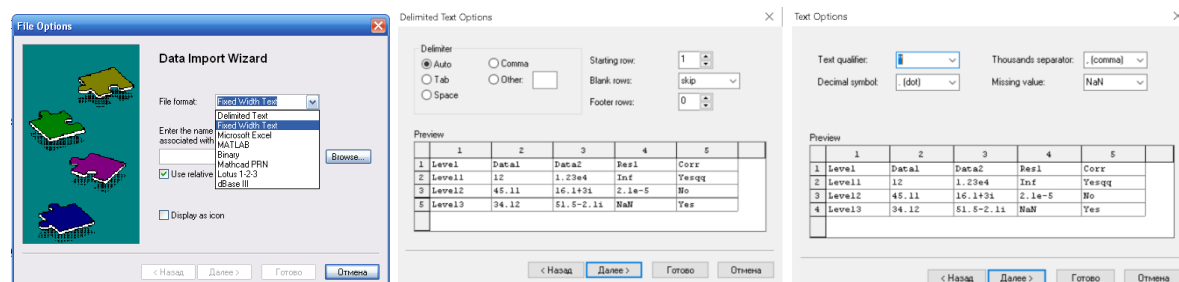
Третє вікно **Text Options** (рис. 3.80 в) містить відображення файлових даних, поля для визначення тексту *Text qualifier*, десяткової точки *Decimal Symbol*, виділення порядків числа *Thousand separator*, заміни нерозпізнаних символів *Missing value*.

Четверте вікно **Data Range** (рис. 3.80 г) містить відображення файлових даних, поля для визначення діапазону рядків для зчитування *Read rows* та діапазону стовпців для зчитування *Read columns*.

По завершенню діалогу в документі *MathCAD* з'являється зображення компоненту у вигляді таблиці з одним маркером (рис. 3.80 д). В цей маркер потрібно ввести ім'я матриці, в яку будуть імпортовані дані.

Пункт контекстного меню компоненту *Properties* відкриває вікно властивостей компоненту, в якому можна редагувати зовнішній вигляд

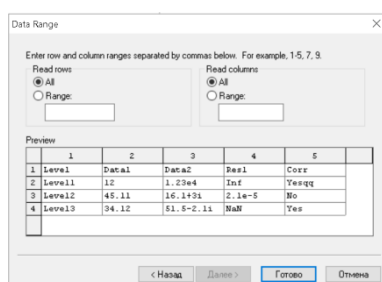
компоненту та параметри, з якими було проведено зчитування. Сторінки вікна в цілому повторюють вікна діалогу універсального імпорту.



а

б

в



г

**д** f :=

	0	1	2	3
0	"Level"	"Data1"	"Data2"	"Res1"
1	"Level1"	12	1.23·10 <sup>4</sup>	"Inf"
2	"Level2"	45.11	16.1+3i	2.1·10 <sup>-5</sup>
3	"Level3"	34.12	51.5-2.1i	...
4				

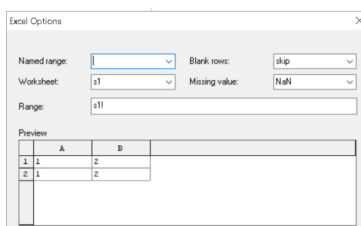
д

Рис. 3.80. Діалогу тексту з розділенням: а – вибір типу файлів; б – вибір розділювача; в – вибір формату тексту; г – вибір діапазону; д – компонент

### Імпорт файлів MS Excel

Друге вікно **Excel Options** (рис. 3.81 а) містить відображення файлових даних, поля для визначення іменованих діапазонів *Named range*, аркуша книги *Worksheet*, діапазону *Range*, дій з порожніми рядками *Blank rows*, заміни нерозпізнаних символів *Missing value*.

По завершенню діалогу в документі *MathCAD* з'являється зображення компоненту у вигляді таблиці з одним маркером (рис. 3.81 б). В цей маркер потрібно ввести ім'я матриці, в яку будуть імпортовані дані.



а

**б** B :=

	0	1
0	2	3
1	2	3
2	12	1

B =  $\begin{pmatrix} 2 & 3 \\ 2 & 3 \\ 12 & 1 \end{pmatrix}$

б

Рис. 3.81. Діалог файлів *MS Excel*: а – вікно діапазону; б – позначка в документі

Пункт контекстного меню компоненту **Properties** відкриває вікно властивостей компоненту, в якому можна редагувати зовнішній вигляд компоненту та параметри, з якими було проведено зчитування. Сторінки вікна в цілому повторюють вікна діалогу універсального імпорту.

### Компонент Excel

Компонент *Excel* вставляється в документ пунктом **Insert-Component** меню.

В першому вікні (рис. 3.82 а) пропонується визначити тип компоненту. Можливо вибрати *Data Import Wizard*, *ODBC Input*, *Matlab*, *Microsoft Excel*, *Scriptable Object*, *Smart Sketch*.

В другому вікні (рис. 3.82 б) можна обрати файл, який буде завантажений в компонент *Excel*.

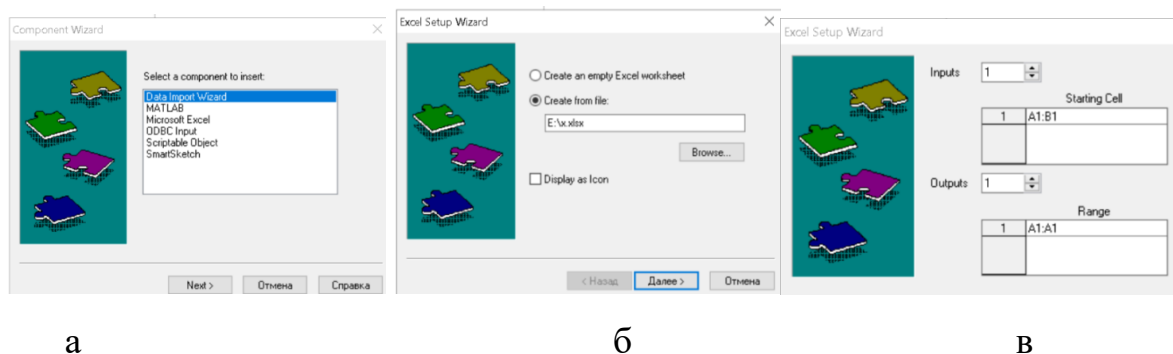


Рис. 3.82. Властивості компонента файл *MS Excel*: а – вибір типу; б – вибір файлу; в – визначення входів та виходів

В третьому вікні визначається кількість входів та виходів та діапазони комірок таблиці *MS Excel*, які будуть відповідати визначеним входам та виходам. Для виходів одна комірка повинна визначатися діапазоном. Аркуші теж повинні визначатися явно.

Компонент *Excel* показується в документі у вигляді таблиці з маркерами входів та виходів (рис. 3.83). Виходи розташовуються в лівому куті згори таблиці. Входи розташовуються під таблицею.

$B := (1 \ 1) \quad A :=$

1	1	3	1
1	2	3	4
1	12	1	2

$A = 1$

B

Рис. 3.83. Компонент *Excel*

Контекстне меню компонента *Excel* дозволяє додати новий або прибрати існуючий вхід/вихід (рис. 3.84а), а пункт *Properties* контекстного меню компонента. Кількість входів/виходів обирається з випадаючого списку *Number of ...*. Призначений входу з номером *K* діапазон відображається у відповідному рядку таблиці *Starting cell*, призначений виходу з номером *K* діапазон відображається в таблиці *Range*.

Подвійний щиклик на рядку таблиці переводить її в режим редагування, в якому можна змінити відповідні значення.

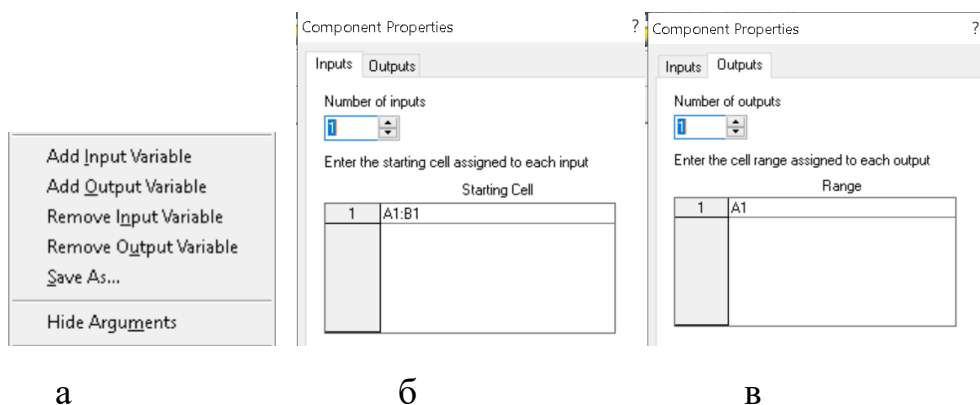


Рис. 3.84. Властивості компонента файл *MS Excel*: а – контекстне меню компонента; б – сторінка входів; в – сторінка виходів

### Засоби *Mathcad Prime*

Передавання даних з *MS Excel* в *Mathcad Prime* може проводитися в «ручному» діалоговому режимі через буфер обміну (*clipboard*) діями копіювання (*Ctrl+C, Copy*) та вставляння (*Ctrl+V, Paste*) між відкритими застосунками *Excel* та *MathCAD*.

Значення скопійованих в *Excel* комірок обраного діапазону імпортуються в *MathCAD* як таблиця. Розмір таблиці визначається автоматично відповідно до обраного діапазону комірок *Excel*. Для

розрахункових комірок імпортується тільки значення без збереження формульної залежності. В якості роздільника між цілою та дробовою частинами дійсного числа в *Excel* має застосовуватися символ крапки, а не коми. В іншому випадку дійсні числа імпортуються в *MathCAD* як рядки.

Для імпорту даних функція **READEXCEL** може бути викликана кнопкою **Input/Output-READEXCEL** стрічки (рис. 3.85 а). При цьому визначення параметрів функції **READEXCEL** відбувається в діалоговому вікні (рис. 3.85 б).

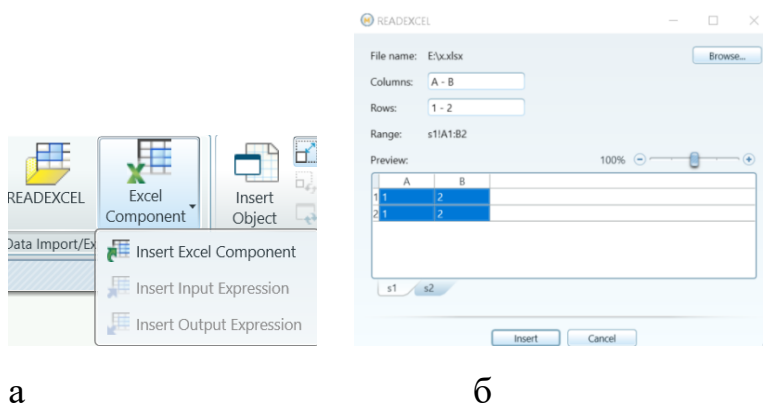


Рис. 3.85. Вікно READEXCEL: а – кнопка виклику; б – вікно параметрів

Вибір файлу для зчитування проводиться стандартним способом натисканням кнопки **Browse**. Ім'я обраного файлу виводиться в полі *File name* вікна. Таблиця з обраного файлу відображається в області *Preview*. Визначення даних для імпорту проводиться позначенням мишею потрібного діапазону комірок в області *Preview* або введенням діапазонів рядків та стовпців в форматі *MS Excel* в полях *Columns* та *Rows*. Індеси комірок розділяються символом тире. Натисканням кнопки **Insert** сформована функція **READEXCEL** вставляється в аркуш *MathCAD*.

## Компонент Excel

### Вставлення компонента Excel в документ MathCAD Prime.

Кожен компонент *Excel* містить три області (рис. 3.86): *входи (Inputs)*, *таблицю*, *виходи (Outputs)*.



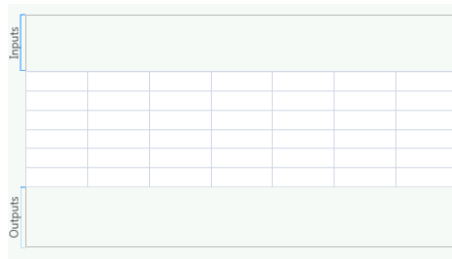


Рис. 3.86. Структура компонента *Excel*

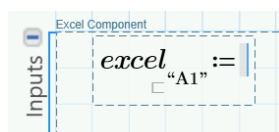
Зображенням областей входів та виходів можна керувати клацанням на піктограмах вгорі на їх ярликах.

Для вставляння компонента *Excel* слід клацнути на місці, куди потрібно вставити компонент *Excel*, та обрати пункт **Input/Output-Excel Component-Insert Excel Component** стрічки.

Введення входів та виходів проводиться шляхом написання виразів прирівнювання у відповідних областях або використанням пунктів **Import/Export-Excel Component-Insert Input Expression/Insert Output Expression** стрічки чи контекстного меню областей входів та виходів. Якщо у відкритому застосунку *MS Excel* перед використанням пункту стрічки виділити діапазон комірок таблиці, то цей діапазон буде вставлений за замовчанням у відповідний вираз компонента *Excel*.

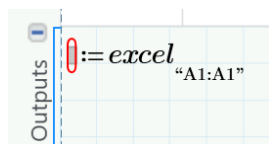
Для введення виразу *вводу* слід клацнути область введення (*Inputs area*) в компоненті *Excel*, ввести індексовану змінну **excel**, в маркер індексу ввести текстовий вираз адреси у форматі *MS Excel* (наприклад, "**Sheet! A1**"), додати оператор прирівнювання (**:=**) та ввести потрібне значення: скаляр, матрицю, рядок або раніше визначену змінну *Mathcad*. Наприклад,  $excel_{\text{"Sheet1!A1"}} := 123$ .

Або обрати пункт **Insert Input Expression** контекстного меню компонента чи пункт **Input/Output-Excel Component-Insert Input Expression** стрічки. *Mathcad* вставляє вхідний вираз за замовчуванням, який слід відредагувати за потребою:



Для призначення значень комірок компоненти *MS Excel* змінним *MathCAD* слід клацнути область *виходів (Outputs area)* в компоненті *Excel*, ввести ім'я змінної введіть знак прирівнювання ( $:=$ ), справа від знаку ввести індексовану змінну **excel** з індексом у вигляді рядка з адресою в форматі *MS Excel*. Наприклад,  $x := excel$  "Sheet1!A1:B2".

Або обрати пункт **Insert Output Expression** контекстного меню компонента чи пункт **Input/Output-Excel Component-Insert Output Expression** стрічки. *Mathcad* вставляє вхідний вираз за замовчуванням, який слід відредагувати за потребою.



Порожні комірки передаються в *Mathcad* як нуль.

Редагувати таблицю компонента можна лише в програмі *MS Excel*. Відкрити *MS Excel* можна подвійним щикликом на компоненті.

Закрити програму *MS Excel* можна пунктом **Close** групи **File** стрічки програми *MS Excel* або подвійним щикликом на таблиці компоненту на робочому аркуші *Mathcad*. При цьому закривається лише книга компонента *Excel*. Інші відкриті книги *MS Excel* залишаються відкритими. При закритті *MS Excel* компонент послідовно виконує вирази входів та вирази виходів.

### 3.7.2. Завдання для самостійного виконання

*Завдання 3.7.1.* Створити матрицю  $4 \times 4$   $M_{i,j} = \frac{i+j}{3}$ . Записати її в текстовий файл з розділенням пробілом. Ширина колонки 6 символів, ширина елементу 3 символи. Зчитати дані в матрицю *A* функцією та інтерактивно файловим обміном. Порівняти результати.

*Завдання 3.7.2.* Створити матрицю  $4 \times 4$   $M_{i,j} = \frac{i+j}{3}$ . Записати її в текстовий файл з розділенням комою в файл *.csv*. Перевірити запис візуально. Зчитати в матрицю *A* дані функцією та інтерактивно файловим обміном. Порівняти результати.

Завдання 3.7.3. Зчитати з файлу *MS Excel* завдання 2.1.3 дані. Побудувати графік.

Завдання 3.7.4. Створити матрицю:

$$\begin{pmatrix} \text{"Level"} & \text{"Data1"} & \text{"Data2"} & \text{"Res1"} & \text{"Corr"} \\ \text{"Level1"} & 12 & 1.23 \times 10^4 & 0.333 & \text{"Yes"} \\ \text{"Level2"} & 45.11 & 16.1 + 3i & 2.1 \times 10^{-5} & \text{"No"} \\ \text{"Level3"} & 34.12 & 51.5 - 2.1i & \text{NaN} & \text{"Yes"} \end{pmatrix}$$

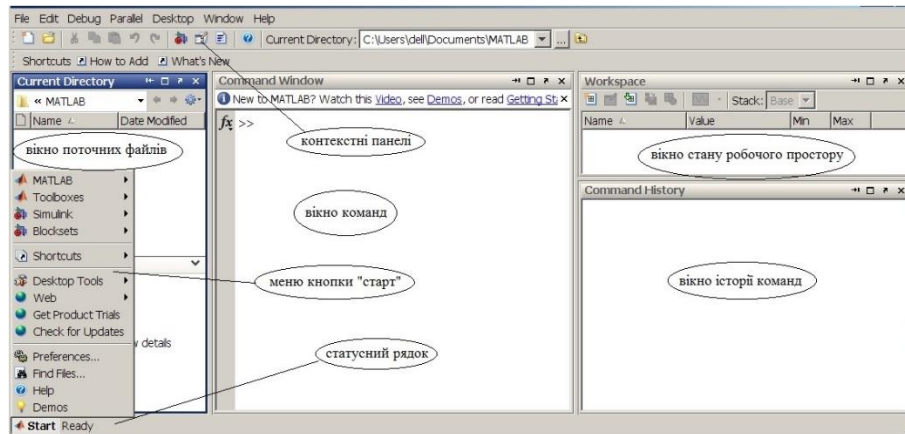
Записати значення в *MS Excel* на аркуш **Data**. Проаналізувати результати.

Завдання 3.7.5. Зчитати дані аркуша «total\_list» завдання 2.2.1. функцією. Побудувати графік аналогічно аркушу 3 завдання 2.2.1.

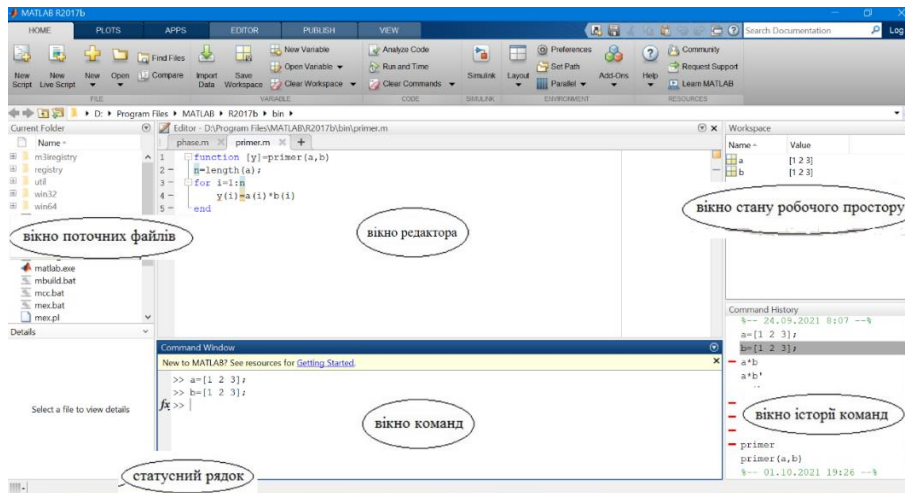
Завдання 3.7.6. Зчитати дані другого аркуша «student\_list» завдання 2.2.1 інтерактивними засобами. Розрахувати статистику аналогічно аркушу 4 завдання 2.2.1.

## 4. СКМ MATLAB

Вікно діалогового середовища *Matlab* є стандартним для *Windows* застосунків. До версії 8.X керування пакетом базувалося на розташованих згори вікна застосунку *меню* та *контекстних кнопкових панелях керування* та розташованій знизу вікна в *статусному рядку* кнопки *Start* (рис. 4.1 а). В подальшому будемо називати версій *Matlab* до 8.X «класичним» з точки зору застосування *меню* та *кнопкових панелей керування*, версії з 8.X – *Matlab*.



а



б

Рис. 4.1 – Оболонка: а – «класичний» *Матлаб*; б – *Матлаб*

В *Matlab* керування пакетом переведено на стрічку *Toolstrips*. Стрічка є аналогом елемента керування *Ribbon* з такою ж українською назвою, який використовується у виробках фірми *Microsoft*. Кнопку **Start** замінено кнопкою керування режимом паралельних обчислень.

В статусному рядку відображається стан системи та параметри редактора. Кнопка «класичного» *Matlab Start* статусного рядка виконувала функції, подібні до функцій однойменної кнопки *Windows*. Вона забезпечує через пункти випадаючого меню швидкий доступ до функцій пакетів розширення *Toolbox []* системи, вікон налаштувань, допомоги тощо (рис. 4.2).

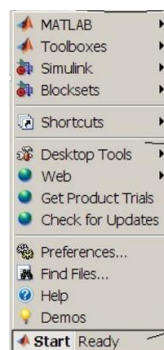


Рис. 4.2. Кнопка **Start**

В центрі оболонки розташовуються *робочі* вікна пакета: *командне вікно*, *вікно історії команд*, *вікно робочого простору*, *вікно поточних файлів*, *вікно редактора*. При виведенні графіків, запуску програм з графічним інтерфейсом додаткові відповідні вікна виводяться в центральну область оболонки.

Вигляд діалогового середовища *Matlab* задається пунктом **Layout** стрічки, «класичного» *Matlab* – пунктом **Desktop** головного меню. Користувач має змогу обрати конфігурацію вікон зі стандартних варіантів *Default*, *Three Column*, *Command window only*, *All but Command minimized*.

*Command window* – командне вікно. Основне вікно для введення команд і відображення результатів розрахунку. В цьому вікні користувачем вводяться з клавіатури команди, відображаються обраховані результати та інформація про помилки.

*Command History* – вікно історії команд. У вікні відображаються по рядках всі уведені команди за час сеансу роботи з пакетом. Клацання на обраному рядку призводить до виконання команди в командному вікні.

*Current directory* – вікно, яке відображає дерево робочих файлів сеансу.

Вигляд та розташування робочих вікон можуть змінюватися в залежності від потреб користувача. Кожне з вікон можна «вийняти» з

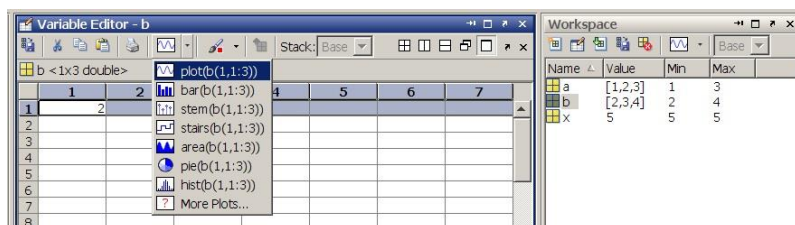
оболонки та зробити вікно незалежним **Undocking**, розкрити на весь екран, закрити, згорнути до значка на краю екрана. Зсув миші з натиснутою лівою кнопкою дозволяє зсунути вікно в інше місце. Керування виглядом робочих вікон здійснюється кнопками керування, які розташовані в правому верхньому куті вікна або відповідними пунктами контекстного меню вікна.

Символами  $\uparrow$  ( $\downarrow$ ) проводиться прокручування рядків введених команд. Виконання команди проводиться натисканням клавіші уведення.

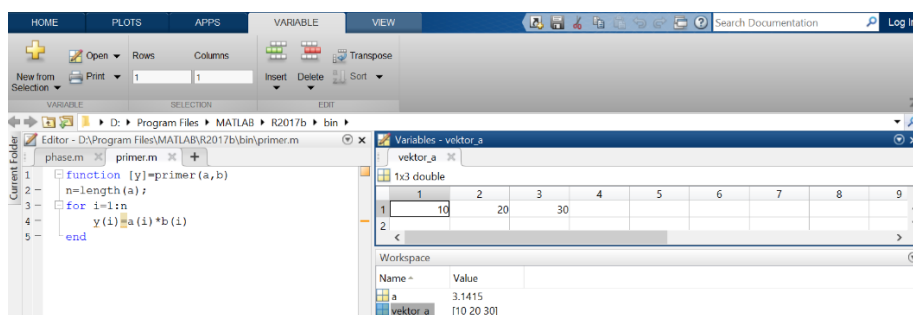
Очистити будь-яке робоче вікно можна пунктом **Clear** відповідного контекстного меню вікна.

Всі данні, які наявні в системі під час сеансу розташовуються в області пам'яті, яку називають *робочим простором*.

*Workspace (робочий простір)* – вікно, яке містить перелік даних та інформацію про зміст даних під час роботи сеансу пакета. Цим вікном слушно користуватися для контролю значень змінних та інших даних. У вікні по рядках відображаються імена, значення, тип даних та об'єм пам'яті, який вони займають (рис. 4.3).



а



б

Рис. 4.3 – Вікно робочого простору: а – «класичний» *Матлаб*; б – *Матлаб*

Наприклад, в сеансі, який зображений на рис. 4.3 б, введено три змінні. Змінна «а» має значенням 3.1415, вектор-рядок «а» має значеннями 10, 20,

30, вектор-стовпець «b» має значення 1, 2, 3. У вікні робочого простору можна змінювати значення даних та перейменовувати змінні. На верхній панелі вікна робочого простору «класичного» *Matlab* знаходяться кнопки для збереження даних у файл, зміни їхнього вигляду та кнопки малювання двомірних графіків.

Подвійний щиглик на якомусь імені робочого простору відкриває вікно редактора даних *Array Editor*. В редакторі даних можна детально передивитись зміст даних, відредагувати існуючі значення, додати нові або прибрати непотрібні. За активації редактора даних в синхронно на стрічці активується закладка **Variable**, яка містить елементи керування редактором.

Дані робочого простору можуть бути збережені для подальшого використання. Збереження даних проводиться в файлах \*.MAT. Збереження проводиться відповідним пунктом контекстного меню робочого простору або в «класичному» *Matlab* пунктом **File-Save Workspace as...** головного меню, в *Matlab* кнопкою **Home-Save Workspace** стрічки.

Завантаження даних робочого простору з дискового файлу може проводитися загальним засобом завантаження файлів **Open** з уточненням типу файла або командою

```
load <ім'я файлу даних> <ім'я 1>,<ім'я 2>, ...
```

в командному вікні.

Для роботи з робочим простором призначені наступні команди:

**clear** очищає робочий простір,

**clear <x>, <y>** очищає обрані дані з пам'яті,

**pack** дефрагментує робочий простір,

**who** показує список даних робочого простору,

**whos** показує список та зміст даних робочого простору.

*Matlab* має два режими роботи: *режим калькулятора* та *програмний режим*. Зовні режими нічим не відрізняються та можуть використовуватися в будь якій послідовності. *Режим калькулятора* передбачає введення в рядку командного вікна виразу для обчислення, команди системи та безпосереднє

їхнє виконання відразу після натискання клавіші «Enter». Програмний режим передбачає створення скрипт-програм або підпрограм користувача в М-файлі з наступним її запуском для виконання. Більшість дій, які виконуються в режимі калькулятора, можливо виконати й в програмному режимі. Для цього потрібно перенести рядки коду в редактор програм та зберегти їх в М-файлі.

## Практикум 4.1. Розрахунки в режимі калькулятора

### 4.1.1. Теоретичні положення

Ознакою того, що *Matlab* готовий до прийняття команди є поява на початку останнього рядка командного вікна знака «»». Для виконання дій достатньо ввести вираз з числами або змінними та натиснути клавішу «Enter». В наступному рядку з'явиться відповідь.

Алгоритми обробки даних складаються з математичних виразів, команд керування, вбудованих та зовнішніх функцій. Вирази можуть вміщувати числа, імена змінних, математичних функцій, операторів дій.

Введення чисел з клавіатури проводиться за загальними правилами мов програмування високого рівня [13] :

- для відділення дробової частини мантиси числа застосовується десяткова точка;
- для чисел в форматі з плаваючою комою десятковий показник задається літерою «e», показник ступеня числа записується як ціле число після символу «e». Між записом мантиси числа та символом «e», який відділяє мантису від показника, не повинно бути символів. Наприклад,  $12,34 \cdot 10^{56} = 12.34e56$ .

Введення інформації в командне вікно підпорядковується таким правилам:

- результат дії відразу виводиться в командне вікно. Якщо оператор закінчується символом «;», то результат його дії не відображається;
- якщо оператор не містить знака «=», тобто є просто записом послідовності дій над числами та змінними, значення результату присвоюється спеціальній системній змінній **ans**. Значення системної змінної **ans** змінюється після дії кожного оператора без знака «=».



Отримане значення змінної **ans** можна використовувати в наступних операторах;

- можна записувати кілька операторів в одному рядку з розділенням комою чи комою з крапкою. Виконання проводиться зліва - направо;
- для продовження запису в кількох рядках наприкінці рядка ставляться три крапки.

Для більш наочного представлення дій та програмування в системі застосовуються змінні. *Matlab* не вимагає попереднього опису типів змінних.

За замовчанням всі чисельні дані обробляються як дійсні числа у вигляді восьмибайтового числа типу *double* у мові C.

Присвоєння має наступний вигляд:

**<ім'я змінної > = <число, рядок або вираз >.**

Ім'я змінної може мати до 30 символів та не повинно співпадати з іменами функцій, процедур системи та системних змінних. Система розрізняє великі та малі літери в змінних. Текстові рядки повинні бути забрані в апострофи.

Система *Matlab* має вбудовані змінні, значення яких зарезервовано:

**i, j** – уявна одиниця (корінь квадратний з -1);

**pi** – число  $\pi$  (3.141592653589793);

**inf** – машинна нескінченність;

**NaN** – невизначений результат ( 0/0 чи inf/inf);

**eps** – похибка операцій над числами;

**ans** – результат останньої операції без знака «="»;

**realmin/intmin** – мінімальна величина дійсного/цілого числа.

З чисел, змінних, функцій в командному рядку набирається обчислювальний вираз. Введення закінчується натисканням клавіші «*Enter*». Обчислення відбувається відразу після закінчення введення.

Наприклад, обчислення наступного виразу

$$0.7854 - \frac{0.7854^3}{1 \cdot 2 \cdot 3} + \frac{0.7854^5}{1 \cdot 2 \cdot 3 \cdot 4 \cdot 5} - \frac{0.7854^7}{1 \cdot 2 \cdot 3 \cdot 4 \cdot 5 \cdot 6 \cdot 7}$$

в командному вікні може мати наступний вигляд

```
>> 0.7854 - (0.7854)^3 / (1*2*3) + 0.7854^5 / (1*2*3*4*5) ...
- (0.7854)^7 / (1*2*3*4*5*6*7) (Enter)
ans = 0.7071
```

Три крапки в кінці рядка дозволяють перенести запис виразу на наступний рядок. Вираз немає знаку рівняння, тому система застосувала системну змінну **ans** та надала їй обчислене значення **0.7071**.

Пакет містить велику бібліотеку вбудованих функцій:

- оператори: +, -, \*, \, /, >, >=, <, <=, =, ==,
- елементарні функції: **sqrt**, **logs**, **log10**, **exp**, **pows** тощо,
- тригонометричні прямі та зворотні функції,
- гіперболічні функції,
- спеціальні функції Беселя, Лагранжа тощо,
- функцій обробки рядків,
- функції роботи з часом,
- функції роботи з матрицями та векторами: **sum**, **prod**, **length**, **size**, **sort**, **ndims**, **max**, **min**, **mean**, **'**, **.**, **rot90**, **dot**, **fliplr**, **cross**, **inv**, **eig**, **det**,
- функції обробки даних та спектральних перетворень,
- функції обробки поліномів,
- функції двомірної та тримірної високорівневої графіки,
- функції низькорівневої дескрипторної графіки.

Виклик вбудованої або функції користувача проводиться наступним чином:

```
<ім'я (імена) результату>=
```

**<ім'я функції>(<перелік фактичних значень аргументів >)**

Функції *Matlab* мають особливість. Вони можуть мати змінну кількість аргументів та результатів. Це значно розширює можливості обчислень. Наприклад, функція **`x=find(A)`** поверне вектор «x» порядкових номерів елементів матриці «A», значення яких не дорівнює нулю. Виклик **`x=find(A,2)`** поверне номери перших двох елементів. Виклик **`[s d]=find(A)`** поверне вектор, перший елемент «s» якого є вектором з номерами рядків елементів, другий «d» - вектор з номерами стовпців.

Аргументи більшості тригонометричних функцій задаються в радіанах. Функції, що приймають аргументи в градусах містять в кінці імені символ «d».

Наприклад, функція **`sin(x)`** потребує аргументу в радіанах, функція **`asin(x)`** повертає результат в радіанах, а функції **`sind(x)`**, **`asind(x)`** працюють з градусами.

Наприклад, обчислення виразу

$$e^{-2.5}(\ln(11.3))^{0.3} + \left(\frac{\sin(2.45\pi) + \cos(3.78\pi)}{\tan(3.3)}\right)^2$$
 має наступний вигляд:

```
>>exp(-2.5)*log(11.3)^0.3+((sin(2.45*pi)...  
+cos(3.78*pi))/tan(3.3))^2  
ans=121.2446
```

Арифметичні та логічні операції виконуються відповідно до пріоритету:

1. Дії в дужках;
2. Функції та логічні дії «так», «або», «ні», «виключаюче або» (and, or, not, xor) тому, що вони є функціями;
3. Логічне заперечення;
4. Транспонування, піднесення до ступеню;
5. Множення, ділення;
6. Складання, віднімання;
7. Логічні відношення: >, >=, <, <=, ==;

8. Логічне «і»: «&»;
9. Логічне «або»: «|»;
10. Логічне «виключаючи або»: «| ».

Дії одного пріоритету виконуються зліва – направо.

Наприклад, два вирази: **and (A, B) + C** та **A&B+C** не еквіваленти.

*Matlab* містить в собі вбудовану арифметику комплексних чисел. Більшість елементарних математичних функцій побудовані таким чином, що аргументи допускають комплексні числа, а результати також формуються як комплексні числа. Для позначення уявної одиниці в *Matlab* зарезервовано два імені «i» та «j».

Введення з клавіатури значення комплексного числа проводиться шляхом запису в командне вікно рядка:

**<ім'я комплексної змінної>=<значення ДЧ>+i\*<значення уЧ> ,**

де ДЧ – дійсна частина комплексного числа, УЧ – уявна частина.

Функції обробки комплексних чисел:

**real (Z)** – виділяє дійсну частину комплексного аргументу Z;

**imag (Z)** – виділяє уявну частину комплексного аргументу Z;

**angle (Z)** – обчислює значення аргументу  $\varphi$  комплексного числа Z (в радіанах від  $-\pi$  до  $+\pi$ )

$$\varphi = \arctg\left(\frac{\text{imag}(Z)}{\text{real}(Z)}\right) ;$$

**conj (Z)** – повертає число, комплексно-спряжене відносно Z;

**complex (A, <B>)** - повертає комплексне число з дійсною частиною A та уявною частиною B. В разі відсутності аргументу «B» повертається комплексне число з нульовою уявною частиною.

$$Z = A + jB = |Z|e^{j\varphi} = \sqrt{\text{imag}(Z)^2 + \text{real}(Z)^2} \cdot \exp\left(j \cdot \arctg\left(\frac{\text{imag}(Z)}{\text{real}(Z)}\right)\right) = |Z|\exp(j \cdot \text{angle}(Z))$$

**Логічні операції**

Результатом логічної операції може бути «істина» чи «брехня». «Істині» в *Matlab* відповідає значення 1, «брехні» – 0.

В системі реалізовані *інтегральна та бітова логіки*. Інтегральні дії застосовуються до значення аргументу. Результатом можуть бути 1 або 0.

Наприклад,

```
» x > 2 ans = 1
```

```
» 2 & 3 ans = 1
```

Бітові операції проводяться над кожним бітом цілих аргументів. Результатом є ціле число.

Наприклад,

```
» a = 3; c = 1;
```

```
» bitand(a,c)
```

```
ans = 2
```

В *Matlab*, як і в *Mathcad*, використовуються логічні відношення (>, <, == тощо), логічні операції (**or**, **and** тощо). Оператори відношення, логічні та арифметичні операції можна поєднувати в одному виразі .

Наприклад, вираз **a + (x > c)** для x=5, c=3 не є помилкою.

Умова **x > c** виконується, результатом є 1. Результат операції додається до значення a.

Результат виконання логічного виразу може бути збережено в змінній:

```
» f = x > c
```

Основні вбудовані функції пакета наведено в додатку А.

Для отримання інформації про можливості та синтаксис вбудованих функцій призначені функції **help**, **lookfor** .

Функція **help fun** виводить в командне вікно опис синтаксису функції **fun** .

Список операторів можна вивести командою **help** з аргументом **ops**, список елементарних функцій – **elfun**, спеціальних – **specfun**, функцій обробки рядків – **strfun**, введення/виведення – **iofun**, роботи з часом -

**timefun**, роботи з матрицями – **matfun**, обробки даних та спектральних перетворень – **datafun**, обробки поліномів – **polyfun**, двомірної графіки – **graph2d**, тримірної графіки – **graph3d**, дескрипторної графіки – **graphics**, графічного редактору – **uitools**.

Функція **lookfor word** проводить пошук слова **word** та виводить список функцій, в яких це слово зустрічається.

### Функції користувача

В разі необхідності проведення в режимі калькулятора обчислень, що мають частини, які повторюються, можна застосувати вбудовану (**Inline**) та анонімну функції користувача. Функції користувача визначаються в командному вікні та викликаються як і всі інші функції *Matlab*. В разі потреби обчислити вираз, який визначено текстовим рядком, слушно застосувати функцію **eval('вираз-рядок')**.

Наприклад, рядки

```
» eval('sin(2)');  
» x=2; eval('sin(x)');
```

дають однакову відповідь – розрахунку функції синуса з аргументом 2

```
ans =    0.9093
```

### Inline-функція

Вбудована функція користувача визначається словом **inline**:

**ім'я\_функції=inline('формула', <'arg1', ..., argM'>)**, де **'формула'** є текстовим рядком з виразом для обчислення. Рядок з переліком аргументів використовується, якщо потрібно змінити порядок аргументів при виклику функції.

Наприклад, для обчислення наступного виразу

$$z = \frac{(\sin(y) + \sin(2y))^3}{1 + \frac{\sin(y) + \sin(2y)}{1 + e^x}} + \sqrt{1 + \frac{\sin(y) + \sin(2y)}{1 + e^x}}$$

потрібно декілька разів визначати значення  $\sin(y) + \sin(2y)$  та  $1 + e^x$ . Можна визначити ці частини у вигляді функцій користувача

```

>>myexp=inline('1+exp(x)')
myexp= Inline function: myexp(x) = 1+exp(x)
>>mysin=inline('sin(y)+sin(2*y)')
mysin=Inline function: mysin(y) = sin(y)+sin(2*y)
>>mysin(2)^3/(1+mysin(2)/myexp(1)). . .
+sqrt(1+mysin(2)/myexp(1))
ans = 1.0237
>> y=2; x=1;
>>mysin(y)^3/(1+mysin(y)/myexp(x)). . .
+sqrt(1+mysin(y)/myexp(x))
ans = 1.0237

```

Якщо функція має кілька аргументів, то вони розташовуються в алфавітному порядку:

```

>> fun1 = inline ('sin(a*x) - x^2*cos(b*x) ')
fun1=inline function:fun1(a,b,x)=sin(a*x) - x^2*cos(b*x)

```

Для зміни порядку аргументів їх слід вставити через кому після рядка-виразу:

```

» fun2=inline('sin(a*x)-x^2*cos(b*x)', 'x', 'a', 'b')
fun2=inline function:fun2(x,a,b)=sin(a*x) - x^2*cos(b*x)

```

### Анонімна функція

Анонімна функція користувача визначається символом «@»:

**Ім'я\_функції = @(список\_аргументів) формула.**

На відміну від **inline**-функції, аргументи та формула записуються в звичайному вигляді, а не як текстовий рядок в лапках. За способом використання анонімна функція подібна до **inline**-функції з тією відмінню, що створюється покажчик на функцію.

```

>>myexp1=@(x) 1+exp(x)
Myexp1= @(x)1+exp(x)

```

```

>> mysin1=@(y) sin(y)+sin(2*y)
Mysin1=@(y) sin(y)+sin(2*y)
>>mysin1(2)^3/(1+mysin1(2)/myexp1(1)) . . .
+sqrt(1+mysin1(2)/myexp1(1))
ans = 1.0237
>> y=2; x=1;
>>mysin1(y)^3/(1+mysin1(y)/myexp1(x)) . . .
+sqrt(1+mysin1(y)/myexp1(x))
ans = 1.0237

```

### Формат виведення результатів

Чисельний результат виводиться системою у вигляді, який визначається попередньо заданим форматом чисел. Цей формат може бути встановлено за допомогою вікна **Preferences** (рис. 4.4) або введенням команди формату явно в рядок командного вікна. Пункт **Command Window** у вікні містить список *Numeric Format*, який визначає формат представлення чисел, що виводяться в командне вікно в процесі розрахунків. Відкрити вікно **Preferences** можна з меню **File**, контекстного меню кнопки **Start** або зі стрічки **Home**.

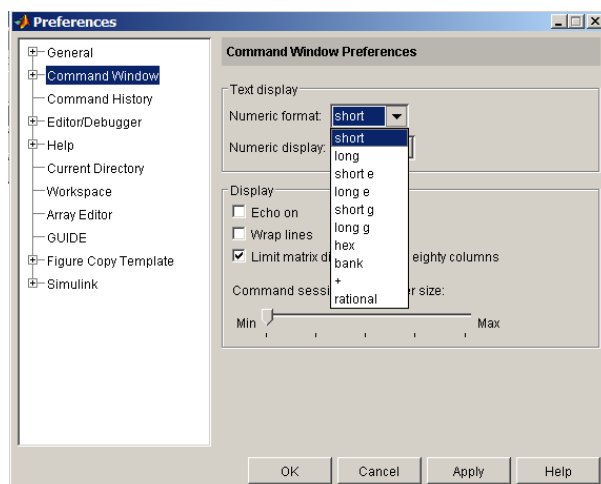


Рис. 4.4. Меню **Preferences**

Нижче наведено найбільш застосовувані формати:

**Short (default)** – короткий запис (застосовується за умовчанням);



- Long** – довгий запис;
- Hex** – запис у вигляді шістнадцятиричного числа;
- Bank** – запис з сотими частинами;
- Plus** – записується тільки знак числа;
- Short E** – короткий запис в форматі з плаваючою комою;
- Long E** – довгий запис в форматі з плаваючою комою;
- Short G** – форма короткого запису в форматі з плаваючою комою (5 значущих цифр);
- Long G** – форма довгого запису в форматі з плаваючою комою;
- Rational** – запис у вигляді раціонального дробу.

Для керування форматом результатів програмно та з командного рядка слугує команда

**format <тип >,**

де можливі значення <тип >:

- short** – запис з фіксованою комою 5 значущих цифр;
- long** – запис фіксованою комою 15 значущих цифр;
- hex** – запис шістнадцятиричного числа;
- bank** – запис з сотими;
- +** – запис тільки знаку числа;
- short e** – запис з плаваючою комою 5 значущих цифр;
- long e** – запис з плаваючою комою 15 значущих цифр;
- short g** – запис з плаваючою / фіксованою комою (5 значущих цифр);
- long g** – запис з плаваючою / фіксованою комою (15 значущих цифр);
- short eng** – запис з плаваючою комою (4 значущих цифри), ступінь кратний 3;

**long eng** – запис з плаваючою комою (16 значущих цифр), ступінь кратний 3;

**rational** – запис у вигляді раціонального дроби;

**compact** – видалення пробілів з запису;

**loose** – додавання пробілів в запис.

Незалежно від формату виведення вихідні дані та результати всіх обчислень в системі *Matlab* зберігаються в пам'яті ПК у вигляді дійсних чисел об'ємом 8 байт з відносною похибкою  $2 \cdot 10^{-16}$  з точними значеннями в 15 десяткових розрядах, діапазон представлення модуля дійсних чисел лягає в проміжку між  $10^{-308}$  та  $10^{+308}$ .

### **Масиви**

*Matlab* є системою, яка спеціально призначена для проведення розрахунків з векторами, матрицями та поліномами.

Термінами *вектор*, *матриця*, *тензор* позначають математичні об'єкти. Термінами *одномірні*, *двомірні*, *багатомірні* масиви – спосіб збереження цих об'єктів в комп'ютері [14].

Вводити масиви в *Matlab* можна:

- повним списком елементів;
- поелементно;
- завантаженням з зовнішніх файлів;
- генеруванням із застосуванням спеціальних вбудованих матричних функцій;

Під вектором в *Matlab* розуміють одномірний масив чисел, під матрицею – двомірний масив, під тензором – багатомірний масив. За умовчанням передбачається, що будь-яка змінна є вектором чи матрицею. Число система сприймає як матрицю розміром (1\*1), а вектор-рядок з N елементами – як матрицю розміром (1\*N).

Звернення до елементів матриць та векторів проводиться вказанням імені та індексу елемента. Індекс записується після імені в дужках з розділювачем у вигляді коми «,». Початковим індексом масивів є 1.

Наприклад,  $\mathbf{a}(1)$  відповідає першому елементу вектора  $\mathbf{a}$ ,  $\mathbf{c}(2,2)$  – елементу з другого рядка та другого стовпця матриці  $\mathbf{c}$ .

Списком елементи вектора задаються через пробіл чи кому в квадратних дужках.

Наприклад, запис рядка  $\mathbf{v} = [1.2 \ -0.3 \ 1.2\mathbf{e}-5]$  задає вектор  $\mathbf{V}$ , з трьома елементами з значеннями 1.2, -0.3,  $1.2 \cdot 10^{-5}$ . Поелементне введення буде мати наступний вигляд:  $\mathbf{v}(1)=1.2$ ,  $\mathbf{v}(2)=-0.3$ ,  $\mathbf{v}(3)=1.2\mathbf{e}-5$ .

Довгий вектор можна вводити частинами, котрі потім об'єднати:

$$\mathbf{v} = [\mathbf{v1} \ \mathbf{v2}]$$

Вектор, елементи якого складають арифметичну прогресію, можна ввести у вигляді, який схожий з ранжованою змінною пакету *MathCAD*. Такий опис змінної називають векторним:

$$\mathbf{I} = \mathbf{nz} : \mathbf{h} : \mathbf{kz} ,$$

де  $\mathbf{nz}$  – початкове значення прогресії (значення першого елемента вектора);  $\mathbf{kz}$  – кінцеве значення прогресії (значення останнього елемента вектора);  $\mathbf{h}$  – крок прогресії. Якщо середній параметр (крок прогресії) не вказано, то він за замовчанням приймається таким, що дорівнює одиниці.

Таким чином вводяться вектори-рядки. Вектор-стовпець вводиться аналогічно, але значення елементів відокремлюються знаком крапки з комою «;».

Функція `linspace( first, last, n)` дозволяє визначити вектор з рівномірно розподіленими елементами першим `first`, останнім елементами `last` та їхньою кількістю `n`. Крок зміни система визначає автоматично.

Функція `logspace(x1, x2, n)` дозволяє визначити вектор з логарифмічно розподіленими `n` елементами в діапазоні від  $10^{x1}$  до  $10^{x2}$ .

Введення значень елементів матриці проводиться в *Matlab* в квадратних дужках по рядках. При цьому елементи рядка матриці один від одного відокремлюються пробілом чи комою, а рядки один від одного - знаком «;» .

Дії над векторами та матрицями проводяться за правилами матричної математики. Для проведення поелементних дій, необхідно перед символом дії поставити символ крапки.

Основні матричні функції пакета наведено в додатку А.

В *Matlab* використано дві невідомі в математиці функції ділення матриць. При цьому вводяться поняття ділення матриць «зліва направо» та ділення матриць «справа наліво». Перша операція записується за допомогою знака « / » , а друга – « \ » [1].

Операція  $\mathbf{B/A}$  еквівалентна послідовності дій  $\mathbf{B \cdot A^{-1}}$ . Її призначено для розв'язання матричного рівняння:  $\mathbf{A \cdot X = B}$ .

Операція  $\mathbf{A \setminus B}$  рівносильна операції  $\mathbf{A^{-1} \cdot B}$ , що є розв'язанням матричного рівняння:  $\mathbf{X \cdot A = B}$ .

Іноді потрібно зробити меншу матрицю з більшої чи, навпаки, вставити меншу матрицю таким чином, щоб вона стала частиною матриці більшого розміру, або провести дії із завданням стовпцем чи рядком. Для подібних дій в *Matlab* застосовується знак двокрапки « : » :

$\mathbf{A(:, n)}$  – виділяє n-ий стовбець з матриці А.

$\mathbf{A(n, :)}$  – виділяє n-ий рядок з матриці А.

$\mathbf{A(n1:n2, m1:m2)}$  – виділяє з матриці А підматрицю, в яку входять рядки з n1 до n2 та стовпці з m1 до m1.

$\mathbf{A(:)}$  – перетворює матрицю в вектор.

Якщо верхньою межею номерів елементів матриці є її розмір, замість числового значення можна використовувати службове слово **end**.

«Розширяти» матрицю можна також, додаванням існуючих матриць («блоків»). Якщо існують кілька матриць-блоків  $\mathbf{A1, A2, \dots, AN}$  з однаковою кількістю рядків, то їх можна об'єднати в одну матрицю А:

$$\mathbf{A = [A1, A2, \dots, AN]}$$

Таку операцію називають *горизонтальною конкатенацією* матриць. Аналогічно, *вертикальна конкатенація* матриць реалізується (за умови, якщо

всі блоки-матриці мають однакову кількість стовпців) шляхом застосування для розділення блоків замість коми з точкою:

$$\mathbf{A} = [\mathbf{A1}; \mathbf{A2}; \dots ; \mathbf{AN}]$$

Для видалення елемента матриці слід присвоїти відповідному елементу «пусте» значення, тобто «[]» – квадратні дужки без пробілу між ними.

Наприклад, для видалення першого рядка з матриці  $\mathbf{A}$  слід записати

$$\mathbf{A}(1, :) = [];$$

Для видалення з матриці  $\mathbf{A}$  другого та третього стовпців слід записати

$$\mathbf{A}(:, 2:3) = [];$$

Примітка. Дії з розширення та скорочення векторів виконуються аналогічно діям з матрицями.

Логічні операції з масивами проходять поелементно. Результатом є масив того ж розміру, що і вихідні.

Наприклад:

$$\gg \mathbf{A} = [1 \ 2 \ 3 \ 4; 5 \ 6 \ 7 \ 8; 9 \ 10 \ 11 \ 12];$$

$$\gg \mathbf{B} = [3 \ 2 \ 3 \ 3; 5 \ 6 \ 2 \ 2; 4 \ 10 \ 11 \ 11]; \quad \mathbf{C} = \mathbf{A} == \mathbf{B}$$

$$\mathbf{C} = 0 \ 1 \ 1 \ 0 \ 1 \ 1 \ 0 \ 0 \ 0 \ 1 \ 1 \ 0$$

*Matlab* зберігає матриці по стовпцях. Для доступу до елементів матриці можна використовувати один індекс, який визначає номер елемента в векторі.

Наприклад, звернення до матриці  $\mathbf{A}(2,2)$  еквівалентно  $\mathbf{A}(6)$ :

$$\gg \mathbf{A}(2,2) \quad \mathbf{ans} = 6$$

$$\gg \mathbf{A}(4) \quad \mathbf{ans} = 6$$

### Масиви комірок

Для зберігання матриць однакових розмірів можна застосовувати тримірні масиви (тензори). Для зберігання різнорідних об'єктів (масивів з різними розмірностями, даних різних типів) призначені масиви комірок. *Масиви комірок* – це багатомірні масиви, елементи яких є копіями інших вже існуючих даних.

Створюються масиви комірок визначенням даних в фігурних дужках.

Наприклад, для матриці  $S$  розміром  $2 \times 2$  виразом

» `C = {S sum(S) min(max(S)) prod(prod(S)) }` отримаємо масив комірок  $C$  з першим елементом `[2x2 double]` - у вигляді самої матриці, другим елементом `[2x2 double]` у вигляді вектора результату функції додавання `sum(S)`, третім елементом `[3]` - числом 3: результатом дії функції `min(max(S))`, четвертим елементом - числом 24: результатом дії функції `prod(prod(S))`.

Порожній масив комірок розміром  $M \times N$  створюється командою `cell(MxN)`.

Для доступу до елементів масивів комірок застосовується індексація ім'ям масиву з індексом, який забрано в фігурні дужки. Наприклад,

```
» S{2}
ans =      4      6
» S{3}
ans =      3
```

## Структури

*Структури* (тип *struct*) також є різновидом багатомірних масивів. Аналогом структури є запис (*record*) в мовах програмування. Елементами структури можуть бути різномірні дані. Доступ до елементів структури здійснюється за ключовим словом, яке вказується після символу «.» після імені структури та визначають поля структури.

Наприклад, скалярну структуру  $S$  з трьома полями `name`, `age`, `grade` можна визначити по одному елементу як

```
»S.name = 'Al Gor'; S.age = 22; S.grade = 'A';
або всі разом функцією
struct('field1',VALUES1,'field2',VALUES2,...)
»S=struct('name','Al Gor','age',22,'grade','A')
```

Додавання елементів структури здійснюється поелементно. Наприклад,

```

»S(2).name = 'Gor Al'; S.age = 33; S.grade = 'C+';
»S(2)=struct('name','Gor Al','age',33,'grade','C+')

```

Для того, щоб дістатися до змісту поля структури слід ввести ім'я структури з індексом в круглих дужках та через крапку ввести назву поля. Наприклад,

```

»x=S(1).age
x=22

```

Для того, щоб дістатися до змісту всіх значень поля структури слід ввести ім'я структури та через крапку назву поля забраними в дужки. Застосування фігурних дужок перетворює поле в масив комірок, квадратних – в вектор. Елементами вектора для чисельних полів будуть числа. Єдиним елементом вектора для текстових полів буде рядок, що є зчепленням всіх рядків полів.

Наприклад,

```

»x={S.age}
x = [22      33]           % <1x2 cell> масив комірок
»x=[S.age]
x = [22 33]              % <1x2 double> вектор
»x={S.name}
x = 'Al Gor'      'Gor Al' % <1x2 cell> масив комірок
»x=[S.name]
x = ['Al Gor Gor Al']   % <1x10 char> вектор

```

## Таблиці

В *Matlab* введено табличний тип даних **table**. Таблиця створюється із змінних типу вектор-стовпець або одномірний масив комірок. Різні стовпці таблиці можуть містити різні типи даних.

Створюється таблиця функцією

```

T=table(var1<, . . . , varN, Name, Value>).

```

Аргументами функції є імена змінних або явно записані дані **var**. Елементи змінних заповнюватимуть відповідні стовпці таблиці. Комірки

таблиці наслідують тип елементів змінних. Комірки таблиці з векторів мають тип елементів векторів, комірки таблиці з масивів комірок мають тип комірки (**cell**). Якщо дані беруться зі змінних, то стовпцям таблиці надаються імена відповідних змінних. Якщо дані задаються явно, то стовпцям надаються імені **Var1**, **Var2** тощо.

Опціональними парами «властивість-значення» можна визначити імена стовпців та рядків таблиці. Імена стовпців визначає властивість **VariableNames**, рядків – **RowNames**. Значення властивостей вводяться як масиви-рядки комірок.

Наприклад,

```
table([10;20], {'A+' ;'B'}, 'VariableNames', {'Age', 'Class'}, ...
'RowNames', {'N1', 'N2'})
```

Доступ до даних може проводитися індексами з використанням круглих дужок та фігурних дужок. Результат індексування з круглими дужками буде мати тип таблиця. Результат індексування в фігурних дужках буде мати тип відповідного елементу таблиці, тобто значенням елементу. Індокси задаються аналогічно масивам. Замість другого індексу може використовуватися ім'я змінної відповідного стовпця.

Наприклад, для таблиці T

```
T= 2x2 table
   Var1   Var2
   ----   ----
   a      1
   b      2
```

індексування в круглих дужках елемента 1,1 покаже

```
T(1, 'Var1')    T(1,1)
ans =  table
   Var1
   ----
   a
```

індексування в фігурних дужках елемента 1,1 покаже

```
T{1, 'Var1'}    T{1,1}
```



```
ans = 'a'
```

Перетворення масивів, комірок, структур в таблицю проводиться функціями `array2table`, `cell2table`, `struct2table` відповідно.

### Особливість логічних дій над масивами

При порівнянні двох скалярів результатом є 1 або 0. Порівняння двох масивів виконується поелементно. Результатом є логічний масив з 1 та 0 того ж розміру, що і вихідні масиви.

При порівнянні скаляру з масивом скаляр порівнюється з кожним елементом масиву. Результатом є логічний масив з 1 та 0 того ж розміру, що і вихідний масив.

Логічні масиви з 0 та 1 не є тотожними до математичних масивів з 0 та 1. Логічні масиви створюються в результаті логічних дій над масивами. Вони можуть бути використані для адресації масивів. З нього витягаються номери елементів, значення яких є 1.

Наприклад,

```
>> y=(6<10)+(7>8)+(5*3==60/4)    %y=1+0+1=2
>> b=[15 6 9 4 11 7 14]; c=[8 20 9 2 19 7 10];
>> d=c>=b                        %d =      0  1  1  0  1  1  0
>> b == c                         %ans =     0  0  1  0  0  1  0
>> r = [8 12 9 4 23 19 10]
>> s=r<=10                        %s =      1  0  1  1  0  0  1
>> t=r(s)                          %t =      8  9  4  10
>> w=r(r<=10)                      %w =      8  9  4  10
```

### 4.1.2. Завдання для самостійного виконання

#### Завдання 4.1.1.

а) Обчислити заданий арифметичний вираз, відформатувати відповідь. Порівняти отриманий результат з наведеним [ 13 ] .

$$\frac{(6.6 - 3\frac{3}{14}) \cdot 5\frac{5}{6}}{(21 - 1,25) : 2,5}$$

2,5

$$\frac{2,625 - \frac{2}{3} \cdot 2\frac{5}{14}}{(3\frac{1}{12} + 4,375) : 19\frac{8}{9}}$$

2,8095

б) Обчислити наведені вирази при заданих значеннях змінних, відформатувати відповідь. Порівняти отриманий результат з наведеним.

$$\frac{\sqrt[3]{\sin(a) + b}}{\lg^2(b)}$$

$$a=3,4^0 \quad b=0.89$$

383.7039

$$\frac{b^3 + \sin^2(a)}{\arccos(b^2) + e^{-\frac{b}{2}}}$$

$$a=83^0 \quad b=0.12$$

239/605

Завдання 4.1.2. Задати значення змінним  $x=0.5$ ,  $y=0.1$  та розрахувати значення змінної  $z$  з використанням **inline** функцій:

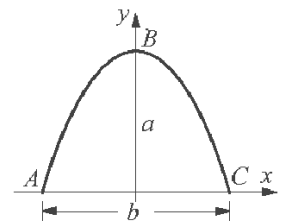
$$\text{а) } z = \arctg\left(\frac{\sqrt[3]{x - \sin(y)}}{\sqrt{1-x^2}} - \frac{|x|\sqrt{1-x^2}}{\sqrt[3]{x - \sin(y)}}\right) \quad \text{б) } z = \frac{\sqrt{\cos(2y) + \sin(4y) + \sqrt{e^x + e^{-x}}}}{(e^x + e^{-x})^3 \cdot (\cos(2y) + \sin(4y) - 2)^2}$$

$$\text{в) } z = \sinh\left(\frac{(x + \ln(y))^3}{\sqrt{|x - \ln(y)|}}\right) \cdot \cosh(x + \ln(y)\sqrt{|x - \ln(y)|})$$

Завдання 4.1.3. Довжина дуги ABC еліпсу з напівосями  $a$ ,  $b$  визначається як

$$L_{ABC} = \frac{\sqrt{b^2 + 16a^2}}{2} + \frac{b^2}{8a} \ln\left(\frac{4a + \sqrt{b^2 + 16a^2}}{b}\right).$$

Розрахувати довжину з використанням анонімної функції для  $a=11$ ,  $b=9$  [15].



Завдання 4.1.4. Ввести довільну матрицю розміром (3\*3) з одним найменшим елементом. Знайти:

а) суму найбільших елементів її рядків;

б) суму елементів рядка, в якому розташовано елемент з найменшим значенням;

в) обернену матрицю. Покажіть, що знайдена матриця є оберненою.

*Завдання 4.1.5.* Ввести дві матриці розміром (2\*2). Побудувати нову матрицю розміром (2\*4) так, щоб перші 2 стовпця були рядками першої матриці, а в інші – стовпцями другої матриці.

*Завдання 4.1.6.* Визначити, в скільки разів змінюється час розрахунку при розрахунку виразу  $\sqrt{|\sin^{0.89}(\sqrt[3]{x})|}$  з використанням «векторизації» та при доступі до елементів векторів через індекс. Діапазон зміни  $x$ : від  $1/e$  до  $10000/e$  з кроком  $1/e$ .

## Практикум 4.2. Програмування

### 4.2.1. Теоретичні положення

Основним режимом роботи з пакетом для проведення обчислювальних експериментів є програмний режим. Програми пакета зберігаються у вигляді *m*-файлів.

В *Matlab* розрізняють два види програм: *Script* (файл-сценарій чи керуюча програма) і *function* (файл-функція). Обидва різновиди мають файлове розширення *. m*.

*Файли-сценарії* призначено для основних програм, які керують обчислювальним процесом. У вигляді *файл-функції* оформляють окремі функції, тобто частини програми, які призначені для багаторазового виклику при змінних значеннях вхідних параметрів і які не можуть бути виконані без попереднього визначення значень вхідних параметрів.

Для написання програм в середовищі *Matlab* призначено вбудований редактор (рис. 4.5). Редактор викликається пунктом *M-file* підменю **New**, пунктом *Open...* підменю **File**.

### Редактор М-файлів

Редактор М-файлів виконує синтаксичну перевірку програмного коду при введенні тексту. При цьому використовуються наступні кольори:

- зарезервовані слова мови програмування - синій колір;
- оператори, константи та змінні - чорний колір;

- коментарі після знаку «%» - зелений колір;
- символічні змінні (в апострофах) - зелений колір;
- синтаксичні помилки – червоний колір.

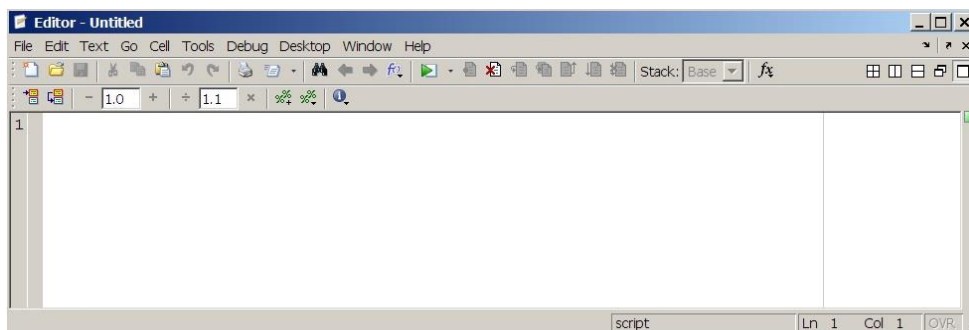










Рис. 4.5. Вікно вбудованого текстового редактора Matlab

-  *Show function* – показ функції;
-  *Set/Clear Breakpoint* – установка/скидання точки переривання;
-  *Clear All Breakpoints* – скидання всіх точок переривання;
-  *Step* – виконання одного кроку програми;
-  *Run* – запуск М-файлу на виконання;
-  *Step In* – покрокове трасування із заходженням в підлеглі М-файли;
-  *Step Out* – покрокова трасування без заходження в підлеглі М-файли;
-  *Exit Debug Mode* – завершення режиму налагодження.

## Налагодження програм

Основним прийомом налагодження М-файлів є установка в них точок переривання (*breakpoints*). Вони установлюються/скидаються кнопкою **Set/Clear Breakpoint**. Скидання всіх точок переривання виконується кнопкою **Clear All Breakpoints**. Для установки точки переривання необхідно розмістити курсор напроти потрібного виразу та натиснути кнопку **Set/Clear Breakpoint**. При запусканні програма буде виконана до

точки переривання, після чого поточні значення змінних будуть виведено у вікні робочого простору *Matlab*.

За допомогою кнопки **Step** можна виконати один крок обчислювань. Жовта стрілка вказує, в якому місці програми стався останов. При трасуванні циклічних дій кожен крок виконання циклу фіксується у вікні командного режиму *Matlab*.

При зупинці в точці переривання можна проконтролювати значення змінних «вручну» (шляхом введення імені змінної в командному рядку ) або за допомогою перегляду точки переривання.

Кнопка **Exit Debug Mode** завершує режим налагодження.

### **Розбиття М-файлу на комірки**

Редактор дозволяє перевірити виконання програми частинами. Для цього використовується розбиття програми на *комірки (Cell)*.

Текст розбивається на *комірки* за допомогою рядків коментарів, які починаються з двох знаків відсотків (%%).

Для запуску засобів редактора роботи з *комірками* слід увімкнути режим *Cell Mode* пунктом **Enable Cell Mode** меню **Cell**. При цьому стають доступними інші пункти цього меню та з'являється панель інструментів для роботи з комірками.

#### Дії пунктів меню **Cell**

*Insert Cell Divider* – вставляння позначки комірок з двох символів відсотка (%%).

*Insert Cell Divider around Selection* – вставляння позначки %% до і після виділеного попередньо фрагменту.

*Insert Text Markup* – вставляння рядків зразків коментарів в місце знаходження курсору.

*Previous Cell, Next Cell* – зсув по коміркам вгору та вниз.

*Evaluate Current Cell* – виконання рядка поточної комірки. Комірка залишається поточною.

*Evaluate Current Cell and Advance* – виконання рядка поточної комірки. Поточною стає наступна комірка.

*Evaluate Entire File* - виконання всього файлу.

Виконання комірок М-файлу проходить без попереднього збереження файлу на диск.

При виконанні файлу потрібно, щоб змінні були доступні в робочому середовищі (*Workspace*).

### **Файли сценарію**

В режимі калькулятора оператор системи вводиться з клавіатури та відразу виконується після натискання клавіші *Enter*. Команди в командному вікні не можуть бути збережені. В режимі калькулятора не можна виконати кілька команд автоматично. Для здолення вказаних особливостей призначені файли-сценарію.

*Файл сценарію* – це записана на диск послідовність команд *Matlab*. При виконанні файлу команди виконуються автоматично рядок за рядком в тій послідовності, в якій вони були записані в файл. Результати дії команд виводяться в командне вікно. Якщо наприкінці рядка в файлі сценарію стоять крапка з комою «;», то результат дії в командне вікно не виводиться.

Запуск сценарію з редактора *Matlab* здійснюється натисканням кнопки **Run**, запуск сценарію з командного вікна здійснюється введенням імені файлу без розширення.

Для того, щоб пакет знаходив файл сценарію, потрібно занести шлях до файлу в список робочого оточення.

Примітка. При завантаженні *Matlab* виконується команда **matlabrc**, яка завантажує для виконання файли **matlabrc.m** та **startup.m**, якщо вони існують.

### **Файл-функції**

Файл-функція має наступну структуру:

**рядок опису;**

рядки коментарю;

тіло функції.

Перший рядок – опис має наступний синтаксис:

```
function <ПВР> = <ім'я процедури>(<ПВВ>),
```

де <ПВР> – перелік вихідних результатів, <ПВВ> – перелік вхідних величин.

Якщо вихідних результатів декілька, вони забираються в квадратні дужки. Елементи переліку вхідних даних та вихідних результатів відокремлюються комами.

Поточні коментарі можуть бути вставлені в будь якому місці файл-функції.

Виклик довідки **help** <ім'я функції> виводить на екран коментарі, які розташовано між рядком опису функції та першим виконавчим оператором чи першим порожнім рядком.

Перший рядок коментарів називається *Hi-line* рядком. Його зміст виводиться командою пошуку функцій **lookfor**, тому рекомендується в цьому рядку вводити коротку інформацію про дію функції.

Всередині файл-функції можуть використовуватись внутрішні файл-функції. При використанні внутрішніх функцій і основна, і внутрішні функції повинні завершуватися службовим словом **end**.

Рекомендується оформляти коментар файл-функції за наступною схемою [16]:

```
function [<ПВР>] = <ім'я функції>(<ПВВ>)
```

```
% Коротке пояснення дії функції.
```

```
% Детальне пояснення змісту, типу, розміру кожної змінної,
```

```
% що вказана в переліку вхідних параметрів ПВВ.
```

```
% Детальне пояснення змісту, типу, розміру кожної змінної,
```

```
% що вказана в переліку вихідних значень ПВР та змінних,
```

```
% які використані як глобальні.
```

```
% Використання інших функцій. Цей розділ заповнюється якщо
```

**%процедура містить звернення до інших функцій, крім  
%вбудованих.**

**% Автор : Вказується розробник, дата розробки.**

**< Порожній рядок >**

Запустити файл-функцію з редактора натисканням кнопки **Run** неможливо. При збереженні файл-функції слід називати файл таким же ім'ям, як ім'я функції.

### **Доступність змінних**

Застосування змінних в файлах-сценаріях та файлах-функціях відрізняється.

В *файлах-функціях* змінні всередині файлу та в заголовку сприймаються як *локальні*, тобто всі значення після завершення процедури витираються та область пам'яті, яка була відведена під ці змінні, вивільняється. Змінні робочого простору та командного вікна всередині *файл-функції* не видимі та навпаки.

В *файлах-сценаріях* всі змінні є *глобальними* та зберігаються весь час сеансу пакета в області пам'яті «робочий простір» (*WorkSpace*). Робочий простір є спільним для всіх Script-файлів та вікна команд.

При необхідності доступу до змінних файл-функції ззовні, вона повинна бути описана як глобальна у всіх файл-функціях та файлі-сценарію, де передбачається її використання

**global <var1> ... <varN>.**

### **Різновиди функцій**

Реалізація алгоритму в одній файл-функції не завжди є оптимальною. Деякі дії доцільно оформити у вигляді окремих службових функцій, які



обслуговують основний алгоритм. Для цього в *Matlab* призначені підфункції (**subfunction**) та вкладені функції.

### Підфункції

*Підфункція* розташовується в тому ж файлі, що і основна функція, після основної функції. Підфункція є залежною від основної функції та може викликатися тільки з основної функції.

Кожне звернення до *підфункції* з основної функції призводить до переходу до операторів *підфункції* та поверненню до точки виклику в основній функції.

*Файл-функція* може містити кілька *підфункцій* зі своїми вхідними і вихідними параметрами. Основна функція може бути тільки одна. Заголовок нової *підфункції* є ознакою кінця попередньої. Основна функція обмінюється інформацією з підфункціями тільки за допомогою вхідних і вихідних параметрів. Змінні в кожній підфункції і в основній функції є локальними, вони доступні тільки в межах своєї функції.

### Вкладені функції

*Вкладені функції* розташовуються в тому ж файлі, що і основна функція безпосередньо в тілі основної функції. Якщо підфункція є зовнішньою по відношенню до основної функції, то вкладена функція є внутрішньою. Змінні основної функції доступні в вкладеній функції.

Основна та вкладена функції завершуються оператором **end**. В одному М-файлі можливо одночасне використання підфункцій та вкладених функцій. Рівень вкладеності функцій не обмежений. Функція може звертатися до своєї вкладеної функції, але не може використовувати вкладену функцію нижнього рівня. Вкладена функція може звертатися до інших вкладених функцій того ж рівня. Функція нижнього рівня може викликати функцію верхнього рівня, в яку вона вкладена.

Змінні основної та вкладених функцій є спільними. Змінні вкладених функцій одного рівня є недосяжними між функціями, тобто є локальними.

### Керування обчислювальним процесом

Для керування обчислювальним процесом система має вбудовані оператори розгалуження, циклу, введення/виведення.

## Оператори розгалуження

Оператори умовного переходу побудовані у вигляді складеного оператора, який починається з одного з службових слів **if**, **switch**, **try** та завершується службовим словом **end**.

В загальному випадку синтаксис *оператора умовного переходу if* має наступний вигляд:

```
if <умова 1>
<операції1 >
elseif <умова 2>
    <операції 2>
...
elseif <умова K>
    <операції K>
...
else
    <операції N> end
```

Рядки **elseif** можуть повторюватися або бути відсутніми, рядки **else** можуть бути відсутніми.

Якщо результат перевірки **умови1** вірний, програма виконує оператори **<операції1>**. В іншому випадку виконується перевірка **elseif**. Якщо умова виконана, виконується відповідна послідовність операторів **<операції K>**. Якщо жодна з умов не є вірною, виконуються **<операції N>**, які вказано після слова **else**.

Перевірка умов завершується після першої істинної умови.

Якщо логічна умова вміщує векторну змінну, то умова буде істиною, якщо вона істинна для ВСІХ елементів змінної.

Примітка. Повна кількість гілок розгалуження оператора **if-elseif-else** дорівнює трьом.

*Оператор перемикання switch* має наступний синтаксис:

```

switch <вираз, скаляр чи рядок символів >
case <значення 1>    <оператори 1>
case <значення 2>    <оператори 2>
...
otherwise            <оператори >
end

```

Оператор реалізує розгалуження обчислювань в залежності від значення змінної чи виразу, шляхом порівняння вказаних значень в рядку **switch** зі значеннями, які вказано в рядках **case**. Відповідна група операторів **case** виконується, якщо значення виразу співпадає з заданим значенням. Якщо значення не співпадає ні з одним із значень в групах **case**, то виконуються оператори, які розташовані за словом **otherwise**.

Вираз в рядку **switch** повинен мати скалярний результат.

Перевірка умов завершується після першої вірної умови.

*Оператор спроби* має наступний синтаксис

```

try
    оператори 1
catch
    оператори 2
end

```

Команда намагається виконати **оператори 1**. Якщо при виконанні виникає помилка або результатом є невизначеність, то будуть виконані **оператори 2**.

### Оператори циклу

Всі оператори циклу побудовані у вигляді складеного оператора, який починається з одного з службових слів **while**, **for** та завершується службовим словом **end**.

В системі реалізовано два різновиди циклу: *цикл з передумовою та арифметичний цикл*. Цикл з передумовою призначено для алгоритмів, в яких

невизначена кількість повторювань. Арифметичний цикл використовується, коли кількість кроків циклу визначена до початку циклу.

Синтаксис оператора *циклу з передумовою* має наступний вигляд:

```
while <умова >  
    <оператори >  
end
```

Оператори всередині циклу виконуються тільки в разі виконання умови. В якості умови можна використовувати логічні або арифметичні вирази. Цикл повторюється доки значення виразу умови не дорівнює нулю.

Синтаксис *арифметичного оператора циклу* має наступний вигляд:

```
for <змінна> = <ПЗ>:<К>:<КЗ>  
    <оператори>  
end ,
```

де <змінна> – ім'я керуючої змінної циклу - лічильника циклу;

<ПЗ> – початкове значення змінної циклу, <К> – значення кроку змінної циклу, <КЗ> – кінцеве значення змінної циклу.

Якщо параметр <К> не вказано, його значення приймається рівним одиниці.

В якості діапазону змінної циклу можна явно вказати весь набір значень у вигляді вектору - рядка чи матриці. Наприклад:

```
1) for m = [ 2, 5, 7, 8, 11, 23 ]  
2) A = [ 1 2; 3 4] ;      for k = A
```

Цикл буде повторюватися стільки разів, скільки стовпців є в матриці А, тобто два рази для наведеного прикладу. В кожній ітерації змінна циклу буде приймати значення чергового стовпця матриці. В такому випадку вона є вектором-стовпцем.

Наприклад, фрагмент

```
S = 0; a = [ 1 2; 3 4] ;  
for k = a ; S=S+sqrt(k(1)^2+k(2)^2); end
```

обчислює суму «довжин» стовпців матриці A.

### Переривання дій

Для дострокового виходу з циклу застосовуються оператори **break**, **continue**. Оператор **break** достроково завершує цикл та передає керування команді, що стоїть після слова кінця циклу **end**. Оператор **continue** завершує виконання операторів, які знаходяться нижче оператора **continue**, та починає наступну ітерацію циклу.

Оператор **return** завершує виконання функції та передає керування в точку виклику – командне вікно, іншу функцію. Схожу дію виконує оператор **error ('message')**. Він також зупиняє виконання програми та виводить в командне вікно текст '**message**' червоним кольором.

### Введення – виведення інформації

Для забезпечення взаємодії з користувачем між M-файлом та командним вікном в системі *Matlab* призначені команди **disp**, **sprintf**, **fprintf**, **input**, **menu**, **keyboard**, **pause**.

Команда **disp(X)** проводить виведення значення вказаної змінної або тексту в рядок командного вікна. Функція може мати тільки один аргумент:

**disp (<змінна або текст в апострофах >)**

Щоб вивести значення кількох змінних за допомогою команди **disp**, потрібно створити єдиний об'єкт, який би містив всі значення. Це можна зробити об'єднанням відповідних змінних в вектор, використовуючи операцію створення вектора-рядка: **x = [x1 x2 ... xN]**.

Тоді виведення значень кількох змінних в один рядок буде мати наступний вигляд:

**disp ([x1 x2 ... xN])**

Аналогічно можна об'єднувати кілька текстових змінних, наприклад:

» **x1='psi'; x2='fi'; x3='teta'; x4='w1';**

» **disp([x1 x2 x3 x4])**

Елементи вектора не можуть мати різний тип. Тобто вони можуть бути або всі числами, або всі - рядками. Для виведення функцією **disp** комбінованої інформації з чисел та тексту потрібно використати функції перетворення чисел в рядки та сформувані єдиний рядок.

Схожу дію виконує оператор **warning('message')**. Він виводить в командне вікно текст **'message'** червоним кольором.

Функція **sprintf** створює текстовий рядок, який містить пояснюючий текст та значення числових змінних у визначеному вигляді. Вигляд рядка задається форматом.

**[S,ERRMSG] = sprintf(format,A, B,...),**

де **A, B,...** – послідовність аргументів. Кожен аргумент може бути скаляром, вектором, матрицею, виразом. Елементи відокремлюються комою. Якщо аргумент **A** – вектор, результуючий рядок формується контактеною всіх елементів;

**format** – опис форми виведення аргументів (дивись додаток A);

**S** – результуючий текстовий рядок;

**ERRMSG** – опціональне поле. Містить повідомлення про помилку або пусту матрицю.

Наприклад, фрагмент

```
x=4; a=0.5; sprintf('x= %i a= %f', x, a)
```

поверне рядок **'x= 4 a= 0.500000'**

```
sprintf('x= %i a= %3.1f', x, a)
```

поверне **'x= 4 a= 0.5'**

```
sprintf('x= %i a= %e', x, a)
```

поверне **'x= 4 a= 5.000000e-001'**

Замість використання для виведення інформації двох функцій **disp**, **sprintf** можна застосувати одну функцію **fprintf**. Ця функція призначена для виведення форматованої інформації в текстові файли.

Функція має наступний синтаксис

```
fprintf(FID,format,A, B,...),
```

Якщо замість дескриптора файлу **FID** вказати в якості першого аргументу 1 або його взагалі не вказувати, то виведення буде проведено на екран.

Функція **input(<запрошення>,<'s'>)** виводить «запрошення» на екран командного вікна та зчитує введене користувачем з клавіатури. Наприклад, **c=input('c=')**.

В разі використання опціонального параметру **'s'**, функція сприймає введене значення як текстовий рядок. Масиви слід вводити в квадратних дужках.

Функція **menu** виводить на екран вікно меню користувача вибору з альтернатив майбутніх дій.

```
k=menu('Перший рядок заголовка', 'Альтернатива1',  
'Альтернатива2', 'Альтернатива n')
```

Виконання програми тимчасово переривається. Система чекає дії у вигляді вибору мишею однієї з кнопок меню з альтернативами. Після вибору функція повертає номер обраної альтернативи (1, 2 чи n). Кількість альтернатив може бути до 32. При зачиненні вікна без вибору альтернативи повертається 0. Для організації циклічного процесу потрібно застосувати разом з командою **menu** команду циклу з передумовою **while**.

Наприклад, запис

```
k=menu('Приклад', 'Вибір 1', 'Вибір 2')
```

призводить до появи на екрані меню, зображення рис. 4.6.

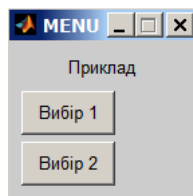


Рис. 4.6. Приклад команди **menu**

Функція **pause (< n >)** тимчасово перериває виконання програми до тих пір, доки користувач не натисне будь яку клавішу

клавіатури. Якщо задано параметр «n», то буде проведено затримку на n секунд.

Функція **keyboard** перериває виконання програми з М-файлу та передає керування клавіатурі (командному вікну). Ознакою передачі керування є символ «k>>».

Користувач може здійснити будь які дії, при цьому йому доступні всі команди системи *Matlab*. Для повернення до виконання програми необхідно набрати команду **return**.

Більш детальний опис функцій введення-виведення наведено в додатку А.

#### 4.2.2. Завдання для самостійного виконання

*Завдання 4.2.1.* [15] Корпус бака складається з циліндру діаметром 24 м, висотою 10 м та конуса висотою 4 м, верхнім діаметром 46 м. Поплавок показує рівень рідини. Написати скрипт-файл, який вводить рівень рідини в метрах, обчислює об'єм рідини в кубічних метрах, виводить результат: «Об'єм рідини=XXX кубічних метрів». Передбачити перевірку введення значення рівня. В разі помилки вивести повідомлення червоним кольором.

Для  $h \leq 10$  м об'єм визначається як  $V = \pi \cdot 122 \cdot h$ . Для  $10 < h \leq 14$  об'єм визначається як  $V = \pi \cdot 122 \cdot 10 + \pi \cdot (h-10) \cdot (122 + 12rh + r_h^2)/3$ , де

$$r_h = 12 + (46-24) \cdot (h-10)/(4 \cdot 2);$$

*Завдання 4.2.2.* Розробити М-функцію перерахунку координат  $X_0, Y_0$  в координати  $X_r, Y_r$  при повороті системи координат на кут  $\varphi$ :

$$X_r = X_0 \cos(\varphi) - Y_0 \sin(\varphi) \quad Y_r = X_0 \sin(\varphi) + Y_0 \cos(\varphi) .$$

*Завдання 4.2.3.* Напишіть М-функції, які обчислюють наближене значення згідно наступних виразів. Аргументом функцій є кількість елементів виразу К.

$$\text{а) } \pi = \sqrt{6 \sum_{n=1}^K \frac{1}{n^2}} \quad \text{б) } \frac{2}{\pi} = \frac{\sqrt{2}}{2} \frac{\sqrt{2+\sqrt{2}}}{2} \frac{\sqrt{2+\sqrt{2+\sqrt{2}}}}{2} \dots$$

*Завдання 4.2.4.* Розробити М-функцію користувача **myrect(x)** для розрахунку «прямокутника» (рис. 4.7):



$$\text{myrect}(U, \tau, \tau_0, x) = \begin{cases} U, & \tau_0 - \frac{\tau}{2} < x < \tau_0 + \frac{\tau}{2} \\ 0, & \tau_0 - \frac{\tau}{2} \geq x \geq \tau_0 + \frac{\tau}{2} \end{cases}$$

Вихідні дані: тривалість імпульсу  $\tau$ , зсув імпульсу  $t_0$ , амплітуда імпульсу  $U$ , точка відліку  $x$ . Результат – значення функції «прямокутника».

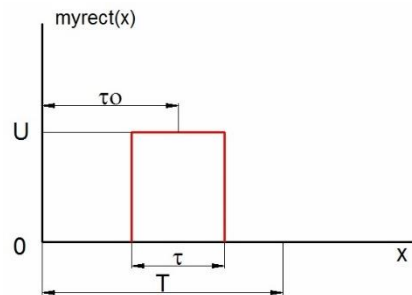


Рис. 4.7. Функція «прямокутника»

Завдання 4.2.5.

а) Розробити М-функцію для розрахунку за формулою Герцбергера:

$$n(\lambda) = A - \frac{B}{\lambda^2 - 0.028} + \frac{C}{(\lambda^2 - 0.028)^2} + D \cdot \lambda^3$$

$A=1.44902$ ,  $B=0.004604$ ,  $C=-0.000381$ ,  $D=-0.0025262$ .

б) Розробити скрипт-файл для розрахунку значень функції п.4.2.5а в діапазоні аргументу від 1 до 2 з кроком 0.1 з виведенням на екран результатів без форматування. Використати векторизацію.

в) Розробити скрипт-файл для розрахунку значень функції п.4.2.5а арифметичним циклом в діапазоні від 1 до 2 з кроком 0.1 з виведенням на екран результатів у вигляді таблиці наступного вигляду:

:	<b>x</b>	:	<b>n(x)</b>	:
:	1.0	:	-	:
		...		
:	2.0	:	-	:

г) Розробити скрипт-файл для розрахунку значень функції п.4.2.5а циклом з передумовою в діапазоні від 1 до 2 з кроком 0.1 з виведенням на екран результатів з форматуванням в наступному вигляді:

**Аргумент:** початок=старт, кінець=стоп, крок=крок, кроків=X

n (1.0) = -

...

n (2.0) = -

*Завдання 4.2.6.* Розробити файл-функцію розв'язання квадратного рівняння та скрипт-файл для введення коефіцієнтів, виклику розробленої функції та виведення відповіді. Скрипт-файл повинен циклічно запитувати значення коефіцієнтів та завершувати роботу при обранні користувачем пункту «Завершити».

1) Вибір користувача проводить рядку командного вікна:

"Подальші дії: Продовжити -1 Завершити -2"

2) Вибір користувача проводиться через меню (рис. 4.8).

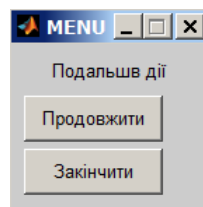


Рис. 4.8. Меню завдання 4.2.6

*Завдання 4.2.7.* Напишіть скрипт-файли, які вводять матрицю та

а) розраховують суму всіх додатних елементів матриці;

б) замінюють в матриці елементи, які більші за середнє значення на середнє значення;

в) визначають максимальний стовповий та рядковий індекси від'ємних елементів матриці.

*Завдання 4.2.8.* Напишіть скрипт, який вводить вектор оцінок від 0 до 100, розраховує та виводить наступну інформацію:

Середній бал=XX

Оцінка А (100-96) XX студентів (XX%)

Оцінка В (86-95) XX студентів (XX%)

Оцінка С (75-85) XX студентів (XX%)

## Практикум 4.3. Двомірні графіки

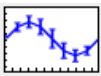
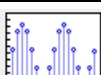

### 4.3.1. Теоретичні положення

Для дослідження функцій однієї змінної, наочного представлення результатів обчислень система дозволяє створити значну кількість типів двомірних графіків за допомогою вбудованих засобів. Основні різновиди двомірних графіків наведено в таблиці 4.1.

Графіки в *Matlab* виводяться в спеціальні окремі графічні вікна **figure** (Рис. 4.9). За замовчанням відкривається одне вікно, в якому відображається один графік. Наступний графік за умовчанням виводиться в те ж вікно замість останнього графіка. При необхідності можливо організувати виведення кількох графіків в одне вікно або відкрити кілька вікон з графіками.

Створення та оформлення графіків може проводитися з командного вікна чи скрипт-файлів, з вікна робочого простору або безпосередньо у вікні графіки.

Таблиця 4.1. Основні типи двомірних графіків

Вигляд	Зміст	Вигляд	Зміст
	Графік функції лініями в декартових координатах в лінійному масштабі		Графік функції лініями з фарбуванням зони під графіком
	Графік функції з діапазоном відхилень лініями в декартових лінійних координатах		Графік функції лініями в декартових координатах в напівлогарифмічному чи логарифмічному масштабі
	Графік функції вертикальними лініями		Графік горизонтальними відрізками ліній
	Стовпова діаграма		Секторна діаграма

	Графік функції в полярних координатах		Векторна діаграма в полярних координатах
	Гістограма в декартових координатах		Гістограма в полярних координатах
	Точкова діаграма		Векторна діаграма в лінійних координатах

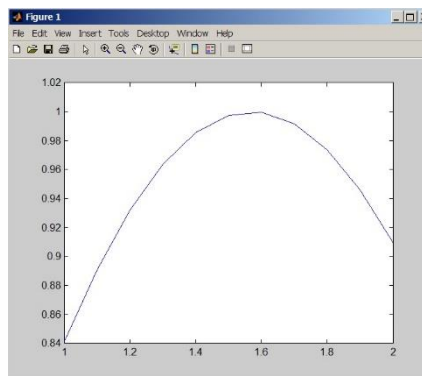


Рис. 4.9. Вигляд графічного вікна

### Базові команди керування графікою

Для створення нового та перемикання між існуючими вікнами використовується команда **figure(<n>)**. При виклику без параметра відкривається нове вікно для графіка. Нове вікно стає поточним. Наступні після створення нового вікна графіки виводяться в поточне вікно. Новому вікну система присвоює номер, який є посиланням на вікно. При виклику з параметром **n** створюється або активується для виведення вікно з номером **n**.

Закрити вікно можна командою **close** або **close(n)**.

Основною функцією графічного відображення векторів є функція

**<N>=plot(<n>,<x1>,y1,<s1>,<x2>,<y2>,<s2>,...)**,

де **xi**, **yi** – вектори, елементами яких є значення аргументу та функції, що відповідають *i*-й кривій графіка; **si** – опціональні символічні формати

графіків. Формати можуть бути задані якостями лінії у вигляді рядка з трьох символів в апострофах 'лінія маркер колір' або послідовності «якість», «значення» в термінах низькорівневого програмування; **n** – опціональний номер вікна, в яке буде виведено графік. В разі відсутності графік виводиться в поточне вікно.

Коди типів ліній: «-» – суцільна, «:» – точкова, «-.» – штрих-пунктирна, «--» – пунктирна.

Коди кольорів: **b** – синій, **g** – зелений, **r** – червоний, **c** – рожевий, **m** – блакитний, **y** – жовтий, **k** – чорний, **w** – білий.

Коди маркерів: «.» – точка, «o» – коло, «x» – хрест, «+» – плюс, «\*» – зірка, «s» – квадрат, «d» – ромб, «v» – трикутник вниз, «^» – трикутник вгору, «<» – трикутник вліво, «>» – три, «p» – п'ятикутник, «h» – шестикутник.

Наприклад, запис

```
>>x=0:0.1:6; y=cos(x);  
>>plot(x, sin(x), '-c', x, y, 'xr')
```

будує в графічному вікні графіки функцій  $f(x)=\sin(x)$  та  $f(x)=\cos(x)$ . Перший графік малюється безперервною лінією блакитним кольором, другий – пунктирною лінією з символом «x» червоним кольором.

Схожу дію виконує команда **area(<X>, Y, <LIMIT>)**. Результатом виклику функції **area** з двома аргументами – векторами X, Y - є графік, аналогічний функції **plot**, тільки з зафарбованою областю під графіком. В разі, коли Y є матрицею, будуються графіки стовпців матриці з накопиченням значень. Опційний параметр **LEVEL** визначає горизонтальну лінію нижньої межі графіка.

Команда **clf** очищує поле графіка.

Команда **<H>=loglog(...)** виводить графік функції в логарифмічному масштабі по обом осям. Синтаксис аналогічний команді **plot**.

Команди **<H>=semilogx(...)**, **<H>=semilogy(...)** виводять графік функції в напівлогарифмічному масштабі по осі X або Y. Синтаксис є аналогічним команді **plot**.

Команда `[AX, H1, H2]=plotyy(X1, Y1, X2, Y2, <F1>, <F2>)` аналогічна функції `plot`, але з відображенням двох вертикальних осей: ліва - для даних  $X1, Y1$ , права - для даних  $X2, Y2$ . `<F1>, <F2>` - опціональні рядки, що забрані в апострофи, вказують тип графіка: `plot`, `semilogx`, `semilogy`, `loglog`, `stem` тощо.

Команда `hold on` вмикає виведення графіків на вже відкрите вікно без витирання раніше намальованих. Команда `hold off` завершує дію `hold on`.

Команда `<H>=polar(T, R, <S>)` - виводить полярний графік залежності  $R(T)$ . Форматується аналогічно функції `plot`.

Команда `[X, Y]=fplot(F, L, <T>, <N>, <S>)` виводить графік функції  $F$  в діапазоні  $L$  з точністю `<T>` типом лінії `<S>` та числом точок `<N>`. Функція  $F$  може задаватися рядком з іменем функції, наприклад, `fplot('myfun(x)', [-1 1])`; посиланням на функцію, наприклад, `fplot(@myfun, [-1 1])`; безпосереднім записом функції, наприклад, `fplot(@x*sin(1./x), [-1 1])`.

Діапазон є вектором зі значеннями `[XMIN XMAX]` чи `[XMIN XMAX YMIN YMAX]`. Тип лінії визначається аналогічно команді `plot`. Чисельне значення  $0 < T < 1$  визначає відносну точність малювання шляхом зміни кількості точок та кроку між ними. Чисельне значення  $N \geq 1$  визначає мінімальну кількість точок на графіку.

Наприклад,

```
>>fplot('1+simn(x)', [0 4], ':', 0.1)
```

малює точками графік виразу `1+simn(x)` з функцією користувача `simn(x)` для значень аргументу від 0 до 4 з точністю 0.1.

Спрощеним аналогом команди `fplot` є команда

```
ezplot(FUNX<, FUNY>, <[TMIN, TMAX]>, <FIG>).
```

Функція малює графік виразу `FUNX` в діапазоні  $-2\pi < X < 2\pi$ . В разі розташування після коми виразу `FUNY` будується параметричний графік `FUNX(T)` та `FUNY(T)`. Опціональний вектор `[TMIN, TMAX]` визначає

діапазон зміни аргументу. При існуванні вектору діапазону значенням посилення (номеру) на вікно **FIG** можна направити виведення графіку в визначене вікно.

### Оформлення графіків

Оформлення графіків проводиться ПІСЛЯ створення вікна та виведення графіків в нього. На полі графіку можна додати сітку, визначити діапазон осей, нанести написи на осі, нанести пояснення на графік.

Функція **grid <mode>** малює сітку на полі графіка. Сітка координат завжди відповідає «цілим» крокам.

Опції можливих режимів:

**on \ off** – включає/виключає відображення основної сітки,

**без параметра** – перемикає стани відображення основної сітки,

**minor** – переключає відображення мілкої сітки.

Функція **axis (<[xmin, xmax, ymin, ymax]>, <mode>)** додає (змінює) розмірні осі графіка. Параметр **<mode>** :

**OFF/ ON** – вмикає/вимикає зображення,

**auto** – встановлює масштаби по обом осям за замовчанням,

**ij** – пересуває початок відліку в лівий верхній кут (матрична система координат),

**xу** – встановлює декартову систему координат з початком відліку в лівому нижньому куту,

**equal** – встановлює однаковий масштаб по обох осях графіка,

**SQUARE** – встановлює квадратну форму поля вікна,

**NORMAL** – поновлює поле графіку до повного розміру.

Без визначення вектора масштабу ключові слова слід писати без дужок. Визначення масштабу потребує написання команди з аргументами в дужках, а ключових слів – в апострофах. Додаткове слово **tight** стягує зображення до меж графіка. Може використовуватися разом з іншими режимами.

За замовчанням діє режим `'auto'`, `'on'`, `'xy'`. Визначення вектора масштабу `[xmin, xmax, ymin, ymax]` встановлює режим `manual`.

Наприклад,

```
>>axis equal tight
```

```
>>axis ([-1 1 2 2])
```

Функція `title('текст')` виводить заголовок над графіком.

Функція `xlabel('текст')` виводить напис осі X.

Функція `ylabel('текст')` виводить напис осі Y.

Функція

```
text(x,y,'текст',<'FontName','Font','FontSize','Size'>)
```

розміщує текст на полі графіка, при цьому початок тексту розміщується в точку с координатами x та y. Координати повинні бути задано в одиницях величин, в яких будується графік та знаходитись всередині діапазону цих величин.

Функція `legend` виводить коментарі у вказане місце.

```
<[L,OB,OU,OT]> = legend <(M)>, де
```

```
M - 'string1','string2',..., 'OFF/SHOW', 'BOXOFF/BOXON',  
'Location', LOC, 'Orientation', OR ;
```

`string1` є поясненням до графіка 1, `string2` до графіка 2 і т.д.;

`OFF/ SHOW` – керує видимістю коментаря;

`BOXOFF/BOXON` – керує відображенням контуру навколо коментаря;

`'Location','LOC'` – визначає положення коментаря;

`'Orientation','vertical/horizontal'` – визначає орієнтацію коментаря.

Функція `subplot(m,n,p)` розбиває область виведення графічної інформації на кілька підобластей, в кожну з яких можна вивести графіки функцій. Діапазони змін змінних по осях координат цих підобластей незалежні одне від одного. Аргументи: m – кількість стовпців, n – кількість



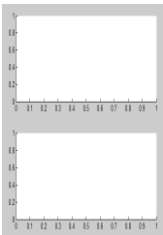
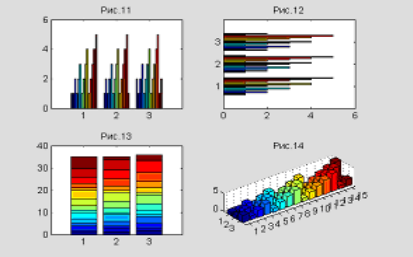
рядків,  $p$  – номер вікна, в якому буде малюватися графік. Підвікна нумеруються зліва направо та згори вниз. Можливо в якості номера підвікна  $p$  використовувати вектор з об'єднаними в одне кількох підвікон.

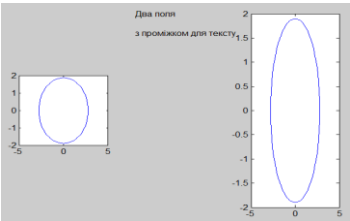
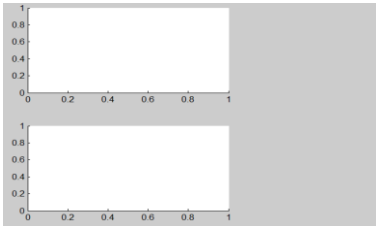
В таблиці 4.2 наведено приклади організації графіків в одному вікні.

Функція `annotation(<fig,>TYPE<,pos>)` виводить на поле графіка вікна `fig` елемент оформлення типу `TYPE` в місце з координатами `pos`. Координати задаються у відносних величинах розміру поля графіка від 0 до 1. Можливо виведення прямокутника `'rectangle'`, еліпса `'ellipse'`, тексту `'textbox'`, лінії `'line'`, стрілки `'arrow'`, двосторонньої стрілки `'doublearrow'`, `'textarrow'` стрілки з текстом.

Опційне поле `pos` визначає вектор для прямокутника, еліпса, тексту: координати  $x, y$  нижнього лівого кута, довжина, висота, – для стрілки, лінії: два вектори  $x, y$  –  $x(1), y(1)$  початку,  $x(2), y(2)$  – кінця. Рядок тексту визначається парою `'String', 'Value'`.

Таблиця 4.2. Приклади виведення графіків в одному вікні.

	
<pre>subplot (2,1,1) subplot (2,1,2)</pre>	<pre>subplot (2,2,1);bar(y) title('Рис.11','FontName','Courier') subplot(2,2,2);barh(y) title('Рис.12','FontName','Courier') subplot(2,2,3); bar(y,'stacked')</pre>

	<pre>title('Рис.13','FontName','Courier') subplot(2,2,4); bar3d(y) title('Рис.14','FontName','Courier')</pre>
Поля для двох графіків одне під одним	Поля для чотирьох графіків у вигляді матриці 2x2 з підписами
	
<pre>subplot(1,3,1) plot(x,y);axis square subplot(1,3,3) plot(x,y) subplot(1,3,2) axis off text(0,1,'Два поля') text(0,0.9,'з проміжком для тексту')</pre>	<pre>subplot(2,3,[1 2]) subplot(2,3,[4 5]) subplot(2,3,[3,6]) axis off</pre>
Поля для рядка з двох графіків та поля для тексту між ними	Поля для двох графіків збільшеної ширини один під одним та поля для тексту праворуч на два рядки

Наприклад,

```
>>rh=annotation('rectangle',[.1 .1 .3 .3]);
>>ah=annotation('arrow',[.9 .5],[.9,.5]);
```

В текстові рядки можливо вводити спеціальні символи за допомогою модифікаторів. Модифікатором слугує символ зворотній слеш «\». Наприклад, літеру  $\pi$  вводять як `\pi`. Коди деяких модифікаторів наведено в

таблиці 4.3. Дія модифікаторів розповсюджується на один символ, до кінця рядку, на групу символів, яка забрана в фігурні дужки.

Таблиця 4.3. Модифікатори символів

Вигляд	Код	Вигляд	Код
→	<code>\rightarrow</code>	text	<code>\rm{text}</code>
text	<code>\bf {text}</code>	text	<code>\fontname{name}</code>
text	<code>\fontsize{size}{text}</code>	text	<code>\it {text}</code>
$x_a$	<code>x_{a}</code>	$x^b$	<code>x^{b}</code>
$\alpha$	<code>\alpha</code>	$\varphi$	<code>\Phi</code>
$\beta$	<code>\beta</code>	$\Delta$	<code>\Delta</code>
$\gamma$	<code>\gamma</code>	$\Gamma$	<code>\Gamma</code>

### Спеціальні графіки

Функція `<H> =bar (<x>, y, <w>, <LINESPEC>)` виводить стовпову діаграму  $y(x)$  з шириною стовпця  $W$ . Якщо  $x$  не задано, аргументом вважається номер елемента вектора «у». **LINESPEC** – задає колір стовпців (один з 'rgbycmkw').

Функція `barh (<x>, y, <w>, <LINESPEC>)` виводить стовпцеву горизонтальну діаграму.

Функція `stem (<x>, y, <w>, <LINESPEC>, 'filled')` виводить «лінійну» діаграму з стовпцями нульової товщини та маркерами на кінці ліній. Формат є аналогічним формату функції `bar`. Додаткове поле `'filled'` замальовує маркери кінця ліній.

Функція `errorbar (X, Y, L, U)` виводить графік з нанесенням діапазону зміни значень  $Y$  від  $U$  (вектор верхніх відхилень) до  $L$  (нижніх). Якщо задано один аргумент, вважається, що діапазон є симетричним. Формат аналогічний функції `plot`.

Функція **hist(Y,<X>)** будує графік гістограми. Відображає кількість елементів вектора Y, які потрапили в піддіпазони, що задані вектором X або скаляром X (кількістю підінтервалів). При відсутності X, приймається розбиття на 10 інтервалів. Повертає розподіл N та центри розподілу X.

Функція **pie(Y,X)** виводить секторну діаграму.

Функція **scatter(X,Y,<S>,<C>,<M>,<filled>)** малює точкову діаграму точок по векторах X,Y розміром S кольором C маркером M. Опціональний параметр 'filled' замальовує маркер. Колір точки визначається назвою або RGB вектором: жовтий: [1 1 0] або 'y', або 'yellow'. Червоний: [1 0 0] або 'r', або 'red'. Зелений: [0 1 0] або 'g', або 'green' тощо.

Наприклад, точкова діаграма зафарбованими точками розміром 60 пікселів синього кольору у вигляді ромба може бути намальована виразом

**scatter(x,y,60,'blue','d','filled') .**

Для зображення векторів та комплексних чисел призначено команди векторних діаграм в полярних координатах **compass(X,<Y>,<LINESPEC>)** та декартових координатах **feather(X,<Y>,<LINESPEC>)** .

Функція **compass** малює вектори, які мають початкову точку (0,0) та кінцеву точку, яка визначена елементами векторів X,Y. Команда **feather** малює вектори, початкові точки яких лежать на осі абсцис, а кінцеві визначені елементами векторів X,Y. Якщо застосовано один комплексний параметр, координати беруться з дійсної та уявної частин. Визначення параметрів ліній в опційному параметрі **LINESPEC** аналогічно команді **plot**.

### Рекомендації по організації графічних результатів обчислювального експерименту

Інженерні графіки повинні мати такий вигляд [13,16], щоб з них легко було зчитувати значення функції при будь-яких значеннях аргументу з

відносною похибкою до кількох процентів. Для цього координатна сітка графіків повинна відповідати цілим числам десяткового розряду.

Інженерна графічна інформація повинна супроводжуватися текстом, з якого повинно бути зрозуміло, що зображено на графіку, яка математична модель застосована, які чисельні значення мають параметри досліджуваного об'єкту.

Додатково, можуть бути наведена інформація про деякі значення обрахованих інтегральних параметрів, назву програми, автора, дату експерименту.

Якщо потрібно порівняти графіки функцій, значення яких відрізняються не більш, ніж на один порядок, графіки доцільно виводити в одному графічному полі, якщо діапазони значень функцій значно розрізняються, функції, що описують подібні фізичні процеси та мають невід'ємні значення, доцільно виводити в одне графічне поле в логарифмічному масштабі. Функції однієї моделі, що мають різну фізичну природу та залежать від одного аргументу, доцільно виводити в одному графічному вікні, але в різних графічних полях. При цьому слід розміщувати графіки один під одним таким чином, щоб однакові значення аргументу на всіх графіках розташовувались на одній вертикалі.

### **Побудова графіків з робочої області**

Якщо попередніми розрахунками сформовано векторні дані, які у вигляді змінних розміщені в робочому середовищі, замість ручного написання команд слушно скористатися кнопкою графіки робочого середовища.

Для цього слід виділити вектори аргументу та функції мишею з затисненою клавішею *Ctrl* та обрати тип графіка з випадаючого списку вікна робочого середовища *Workspace* (рис. 4.10). Система сформує відповідну команду графіки та виконає її. Робоче середовище формує команду з використанням засобів дескрипторної графіки.

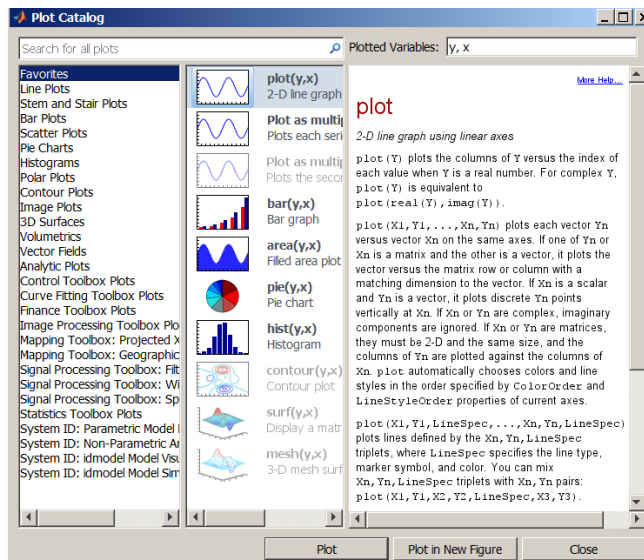


Рис. 4.10. Список графіків робочого середовища

## Побудова та редагування графіків в графічному вікні

Графічне вікно (рис. 4.8) не тільки пасивно відображає графічну інформацію. Воно має потужні інструменти для налаштування вигляду зображення. З включенням всіх засобів вікно має наступний вигляд (рис. 4.11.).

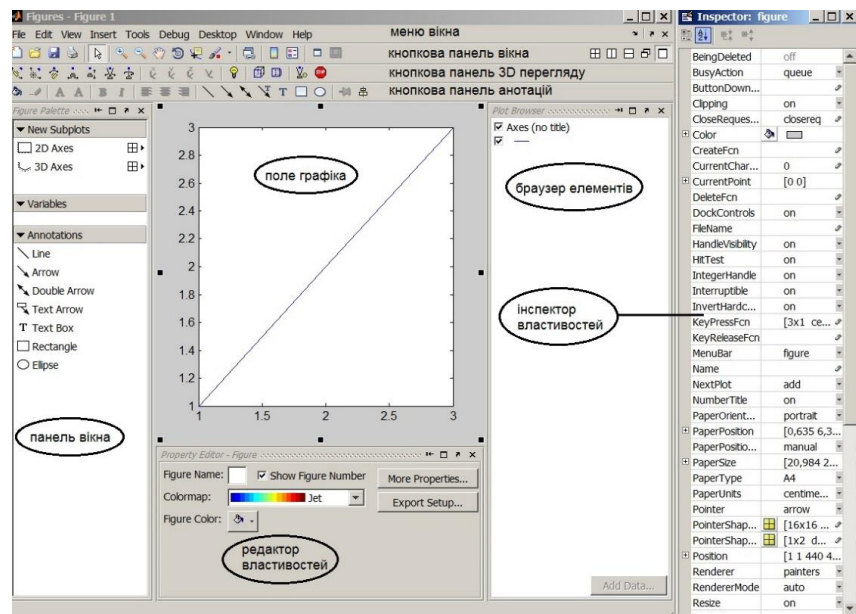


Рис. 4.11. Графічне вікно

Вікно містить поле графіка, меню вікна, кнопкові панелі: вікна **Figure Toolbar**, камери 3D перегляду **Camera Toolbar**, анотацій **Plot Toolbar**, - редактор властивостей графічних елементів **Property**

**editor**, панель вікна **Figure palette**, браузер графічних елементів **Plot browser**, інспектор властивостей графічних елементів **Inspector**.

Меню графічного вікна містить команди для керування самим вікном та інструментами налаштування вигляду графіків. Команди поєднані в логічні групи.


З групи **Edit** опцією **Figure properties** активується редактор властивостей **Property editor** в режимі редагування вікна **figure**, опцією **Axis properties** – в режимі редагування поля графіка **axis**, опцією **Current object properties** – в режимі редагування елемента графіка, який обере користувач щикликом миші на елементі. Опція **Colormap** активує редактор карти кольорів **Colormap editor**, яка використовується для розфарбування тримірних графіків.

Група **View** керує виведенням на екран кнопоквих панелей, редактора властивостей графічних елементів, панелі вікна, браузера графічних елементів.

Група **Insert** повторює кнопкову панель анотацій та панель вікна. В ній розміщені пункти керування вставлянням на поле графіка елементів оформлення: підписів осей **X Label**, **Y Label**, **Z Label**, підпису графіка **Title**, пояснень **Legend**, карти кольорів **Colorbar**, додаткових елементів оформлення у вигляді стрілок, ліній, прямокутників, еліпсів, текстів.

Група **Tool** повторює кнопкову панель вікна та керує опціями перегляду графіка: масштабуванням, зсувом, поворотом.

Групи **Debug**, **Desktop**, **Window**, **Help** повторюють пункти меню системи.

Зміна властивостей графіків може проводитись безпосередньо в графічному вікні. Виклик редактора проводиться через пункт меню **Edit** або кнопкою вмикання режиму редагування  з панелі вікна, командою **plottools** командного вікна. Вигляд та можливості редактора визначаються типом графічного елемента.

На рис. 4.12 показано вигляд редактора для редагування властивостей осей: написів на осях, кольору та шрифту написів, типу осі (лінійна, логарифмічна), діапазоном аргументів осі, виведенням сітки.

На рис. 4.13 показано вигляд редактора для редагування властивостей графіків: типу, кольору, товщини ліній, типу, кольору маркерів. В полях **Data source** можливо змінити зміст з списку визначених в *Workspace* даних по осях.

Для додавання елементів оформлення: підписів осей та графіка, пояснень, карти кольорів, додаткових елементів оформлення у вигляді стрілок, ліній, прямокутників, еліпсів, текстів, - можна використати групу **Insert** меню вікна, кнопки панелі анотації, панель вікна **Figure Palette**. Розміщення елементів та визначення їхніх розмірів проводиться безпосередньо на полі графіка в діалоговому режимі за допомогою миші.

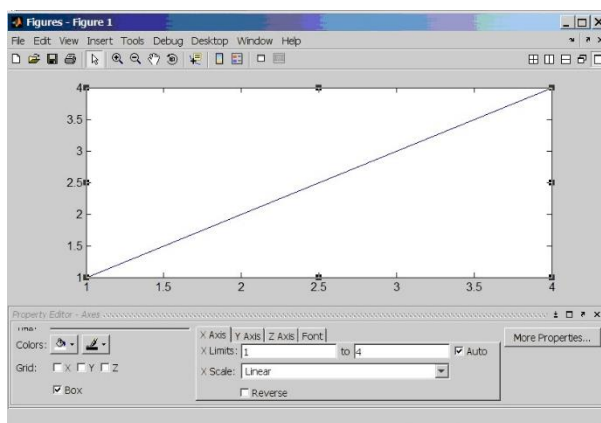


Рис. 4.12. Вигляд редактора в режимі редагування осей

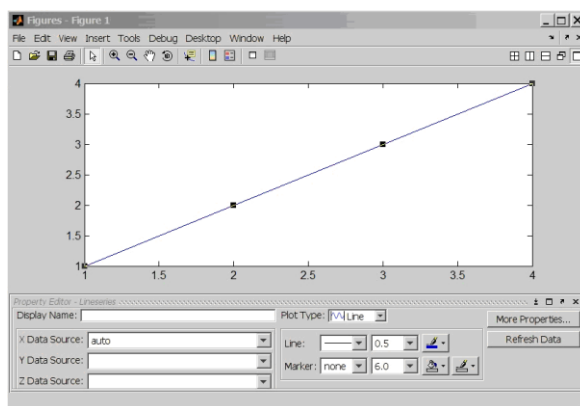


Рис. 4.13. Вигляд редактора в режимі редагування ліній

Зручною є можливість виконання дій з графіками за допомогою контекстних меню. Контекстне меню викликається натисканням правої



кнопки миші після того, як об'єкт обрано клацанням лівою кнопкою миші. Ознакою обрання об'єкта є поява чорних прямокутних маркерів на ньому. Зміст пунктів меню залежить від типу об'єкта (рис. 4.14).

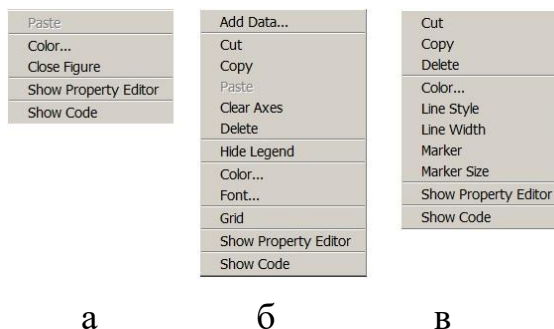


Рис. 4.14. Контекстні меню: а – вікна, б – поля графіка, в – графіка

Інтерактивно керувати кількістю графіків можна за допомогою браузера друку. Простим клацанням миші можна прибрати з поля графіки або кнопкою **Add Data** додати.

### 4.3.2. Завдання для самостійного виконання

*Завдання 4.3.1.* За допомогою функції **fplot** намалювати графік функції  $f(x) = \frac{x^2}{2 + \sin(x) + x^2}$  для  $-4 < x < 4$ .

*Завдання 4.3.2.* На одному полі намалювати графіки функції  $f(x) = x^3 - 2x^2 - 9\sin^2(x) - e^{0.4x}$  та її похідної в діапазоні  $-2 < x < 3$ . Графік функції – червоний суцільною лінією, похідної – синьою штриховою. Додати легенду.

*Завдання 4.3.3.* Намалювати в одному вікні графіки функції  $y(x) = 2^{10-0.5x}$  в діапазоні  $0.1 < x < 1000$  в лінійному, напівлогарифмічному, логарифмічному масштабі. Графіки підписати. Зобразити сітку.

*Завдання 4.3.4.* Для значень часу  $10 < a < 20$  секунд, відповідних значень функції відстані  $y = (920\ 650\ 420\ 335\ 221\ 146)$  метрів та похибки  $d = (14\ 20\ 25\ 12\ 10\ 11)$  метрів намалювати графік. Похибка симетрична. Осі підписати.

*Завдання 4.3.5.* Намалювати в полярних координатах графік функції  $r(x) = 2.5 \cos^2(\frac{x}{3}) + x$  для 150 точок в діапазоні  $0 < x < 2\pi$ .

*Завдання 4.3.6.* [15] Закон ідеального газу пов'язує тиск  $P$ , об'єм  $V$  та температуру  $\frac{T}{PV} = nRT$ , де  $n$  – кількість речовини в молях,  $R = 8.3245$  Дж/(К·моль).

Побудувати на одному полі ізотермічні графіки залежності тиску в Паскалях від об'єму для одного моля газу при  $T = 100, 200, 300, 400$  К. Діапазон зміни об'єму від  $1 \text{ м}^3$  до  $10 \text{ м}^3$ . Підписати графік, осі. Вивести легенду.

*Завдання 4.3.7.*

а) Намалювати графік циклоїди  $x = 12 \cos(t) - 2 \cos(6t)$   $y = 12 \sin(t) - 2 \sin(6t)$  для  $-0 < t < 4\pi$ .

б) Намалювати графік  $x = (3.3 - 0.4t^2) \sin(t)$   $y = (2.5 - 0.3t^2) \cos(t)$  для  $-2\pi < t < 2\pi$ .

*Завдання 4.3.8.* Розробіть скрипт-файл для побудови графіків коефіцієнта передачі амплітуди та фази електричного ланцюга в напівлогарифмічному масштабі один під другим в одному вікні (рис. 4.15) згідно наступного виразу

$$K(\omega) = \frac{n}{1 + a_1 + a_2 - \omega^2 \frac{\tau_1 \tau_2}{a_1} + j\omega(\tau_1 + \tau_2) + \frac{1}{j\omega\tau_3}}$$

де  $n$  – коефіцієнт передачі трансформатора;  $R_f$  – опір перетворювача;  $R_t$  – опір навантаження;  $C_t$  – ємність;  $L_1$  – індуктивність первинної обмотки;  $L_2$  – індуктивність вторинної обмотки трансформатора.

$$a_1 = \frac{R_f}{R_t} \quad a_2 = \frac{L_1}{L_2} \quad \tau_1 = C_t R_f \quad \tau_2 = \frac{L_1}{R_t} \quad \tau_3 = \frac{L_2}{R_f} \quad n = \frac{L_2}{L_1}$$

Вихідні данні задати присвоєнням. Текстові коментарі повинні змінюватися автоматично при зміні вихідних даних. Графіки амплітудної та фазової характеристик намалювати в «напівлогарифмічному» масштабі.

Для форматування вікна графіків рекомендується застосувати розбиття вікна командою **subplot**. В текстовій частині сховати поле графіка командою **axis**, коментарі вивести командою **text**. Рядок тексту сформувати командою **sprintf**. Аргументи графіків доцільно формувати командою **logspace**.

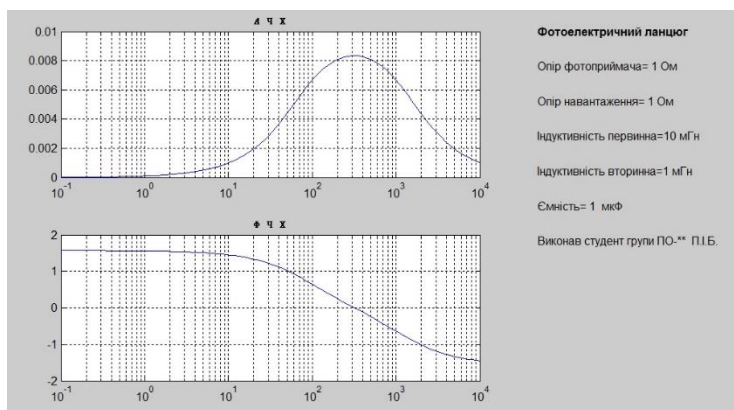


Рис. 4.15. Вигляд вікна завдання 4.3.8

Завдання 4.3.9. Розробити функцію розрахунку  $\cos(x)$  за формулою

Тейлора 
$$csm(x,n) = \sum_{i=0}^n (-1)^i \frac{x^{2i}}{(2i)!}$$
. Для діапазону аргументу  $-4 < x < 4$

сформувати вектори: функцій  $\cos(x)$ ,  $csm(x,2)$ ,  $csm(x,3)$ ,  $csm(x,4)$ . За допомогою засобів графічного вікна сформувати зображення відповідно до рис. 4.16 .

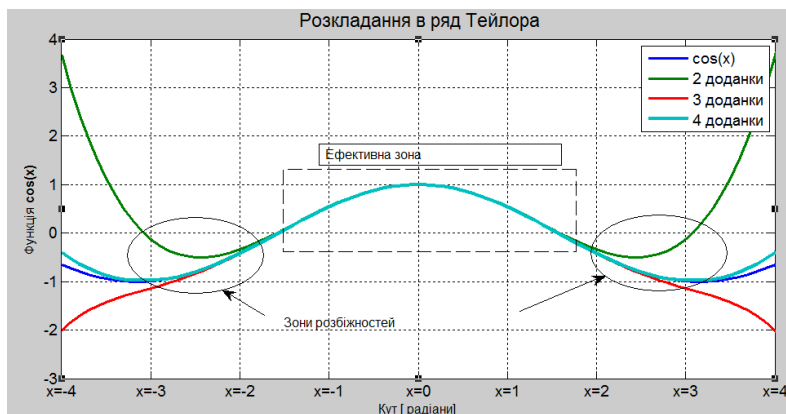


Рис. 4.16. Зображення завдання 4.3.9

## Практикум 4.4. Тримірні графіки

### 4.4.1. Теоретичні положення

Для візуалізації багатовимірних моделей, аналізу матричних даних, функцій двох змінних в *Matlabi* використовуються засоби тримірної графіки. Базові типи тримірних зображень наведено в таблиці 4.4.

Таблиця 4.4. Базові види просторових зображень

Вигляд	Тип	Вигляд	Тип
	Зображення матриці у вигляді поверхні		Зображення матриці у вигляді каркасної поверхні
	Вид згори матриці у вигляді поверхні з ізоліній		Зображення матриці у вигляді поверхні з сіткою по ізолініях
	Зображення просторових кривих		Зображення матриці у вигляді поверхні з сіткою по рядках чи стовпцях

Положення точки в просторі визначається трьома координатами. Наприклад, в декартовій системі координат –  $x$ ,  $y$ ,  $z$ . Сукупність точок, які належать до просторової кривої визначається трьома векторами, які містять координати точок по відповідних координатах. Тримірні поверхні описуються функцією двох змінних та потребують визначення двомірної сітки, в вузлах якої буде розраховуватися значення функції. Для опису такої сітки в *Matlab* використовуються дві матриці. Кількість рядків цих матриць дорівнює кількості вузлів розбиття відрізка  $[y_{\min} y_{\max}]$ , а кількість стовпців - кількості вузлів розбиття відрізка  $[x_{\min} x_{\max}]$ . Рядки матриці  $X$  складаються з однакових рядків, що містять координати вузлів на відрізку  $[x_{\min} x_{\max}]$ , а матриця  $Y$  - з однакових стовпців з координат вузлів відрізка  $[y_{\min} y_{\max}]$ .

Для перетворення векторів  $x$ ,  $y$  в матриці  $X$ ,  $Y$  призначена функція  $[X, Y] = \text{meshgrid}(x, y)$ .

Рядки матриці  $X$  є копіями вектора  $x$ , стовпці матриці  $Y$  є копіями вектора  $y$ .

### Графіки просторових кривих

Для побудови графіків просторових кривих слугує функція

**plot3(x,y,z,<linestyle>)** .

Аргументами функції є вектори  $x, y, z$ , елементами яких є координати точок кривої. Зазвичай вектори містять результати розрахунку параметричного опису кривої  $x=f(t)$ ,  $y=g(t)$ ,  $z=s(t)$ . Можна замість трьох векторів визначити матрицю, в трьох стовпцях якої розмістити вектори  $x, y, z$ . Опціональний параметр **linestyle** визначає тип лінії кривої аналогічно функції **plot**.

Примітка. Функцію **plot3** можна використовувати і для отримання зображення функції двох змінних у вигляді сітчастої поверхні. При цьому в якості аргументів мають бути застосовані матриці сітки  $X, Y$  та значень функції  $Z$ . Малюються лінії паралельно площині  $YOZ$ . Кольори ліній змінюються послідовно відповідно списку кольорів команди **plot**.

### Графіки просторових поверхонь

Для створення зображень поверхонь призначені функції **mesh**, **meshc**, **meshz** та **surf**, **surfz**, **surfl**. Функції **mesh** будують зображення сітчастої поверхні, функції **surf** – поверхні у вигляді оболонки. Колір точки поверхні змінюється відповідно до значення функції в цій точці.

Синтаксис всіх функцій однаковий: **mesh(X,Y,Z,<C>)** .

Аргументами функцій є дві матриці  $X, Y$  сітки та матриця  $Z$  значень функції. Опціональний параметр  $C$  визначає тип фарбування поверхні. За замовчанням  $C=Z$ . Параметр повинен бути матрицею з розміром, який дорівнює розміру матриць інших аргументів.

Функція **mesh** малює сітчасту поверхню з ребрами по двох координатах. Ребра розфарбовуються відповідно значенню функції.

Функція **meshz** додатково заштриховує поверхню в площині  $ZY$  для першого значення  $x_{\min}$  та останнього –  $x_{\max}$ .

Функція **meshc** додатково малює карту ізоліній під зображенням поверхні.

Функція **surf** малює оболонку-поверхню з ребрами та гранями по двох координатах. Грані розфарбовуються відповідно значенню функції.

Функція **surf1** додає джерело освітлення для фотореалістичного ефекту.

Функція **surfc** додатково малює карту ізоліній під зображенням поверхні.

### Контурні графіки

Для кількісного аналізу графіків поверхонь корисно мати наочну інформацію про розподіл значень функції. Таку інформацію містять контурні графіки. На контурних графіках зображуються лінії, в яких функція має однакове значення – ізолінії.

В *Matlab* контурні графіки реалізовані функціями типу **contourX**.

Функція **[c h]=contour(X,Y,Z,<N>,<linestyle>)** зображує двомірну карту ізоліній поверхні. Вона є видом згори на поверхню **Z** у вузлах сітки **X**, **Y**. При виклику без опційного параметра **N** система сама визначає кількість ізоліній. Застосування параметра натуральним числом визначає графік з кількістю ізоліній **N**. Застосування параметра вектора визначає значення, для яких будуть намальовані ізолінії. Кількість ізоліній буде дорівнювати **length(N)**. Якщо визначити вектор як **[v v]**, то буде намальовано одну ізолінію для значення **v**. Опціональний параметр **linestyle** визначає колір, тип ізоліній аналогічно функції **plot**.

Для того, щоб додати на графік кількісні значення ізоліній слід викликати функцію **clabel(c,h)** з результатами виконання функції **contour** з матрицею **c** та вектором **h** в якості аргументів.

Функція **[c h]=contourf(X,Y,Z,<N>,<linestyle>)** малює контурний графік та зафарбовує зони між ізолініями відповідним кольором.

Функція `[c h]=contour3(X,Y,Z,<N>,<linestyle>)` малює аксонометричне зображення ізоліній. Значення `Z` відповідає висоті, на якій зображується ізолінія.

### Оформлення тримірних графіків

Палітра фарбування граней поверхні визначається функцією `colormap('mode')`. Параметр-рядок `'mode'` визначає передумовлені палітри:

- з плавною зміною кольорів: `'hsv'` – червоний-жовтий-зелений-синій-червоний, `'hot'` – чорний-червоний-жовтий-білий, `'gray'` – відтінки сірого, `'bone'` – відтінки синьо-сірого, `'copper'` – відтінки мідного, `'pink'` – відтінки рожевого, `'jet'` – від синього до червоного, `'cool'` – від фіолетового до рожевого, `'autumn'` – від червоного до жовтого, `'spring'` – від рожевого до жовтого, `'winter'` – від синього до зеленого, `'summer'` – від зеленого до жовтого.
- з різкою зміною кольорів: `'flag'` – червоний, білий, синій, чорний, `'lines'` – відповідно списку кольорів ліній в `plot`, `'vga'` – Windows 16 кольорів, `'prism'`, `'white'` – білий.

Функція `colobar` виводить стовпець з поточною палітрою у вікно. Опціонально в якості параметра можна визначити положення палітри `'North'`, `'South'`, `'East'`, `'West'`, `'NorthOutside'`, `'SouthOutside'`, `'EastOutside'`, `'WestOutside'` або вимкнути зображення палітри: `'off'`, `'hide'`.

Для сітчастих поверхонь (`plot3, mesh`) можна командою `hidden on` вимкнути невидимі ребра, тобто з каркасної моделі зробити непрозору оболонку. Команда `hidden off` повертає каркасну модель.

Виглядом видимих ребер керує команда **shading mode**. Значення параметра **'interp'** прибирає ребра та проводить плавне фарбування, значення **'flat'** прибирає ребра та рівномірно фарбує грані, значення **'faceted'** виводить ребра та рівномірно фарбує грані. Є значенням за замовчанням.

Для оформлення графіків поверхонь також можна застосувати функції **text, xlabel, ylabel, zlabel, title, axis, subplot**.

Тримірні графіки зображуються в аксонометричній проекції. Вигляд поверхні залежить від взаємної орієнтації поверхні та точки зору спостерігача. В *Matlab* точка зору визначається кутами азимута  $Az$  та піднесення  $Ei$  (рис. 4.16). Кут азимута рахується від осі  $Y$  в площині  $XOY$  проти годинникової стрілки. Кут піднесення – кут між лінією, яка поєднує точку зору з початком системи координат, та площиною  $XOY$ .

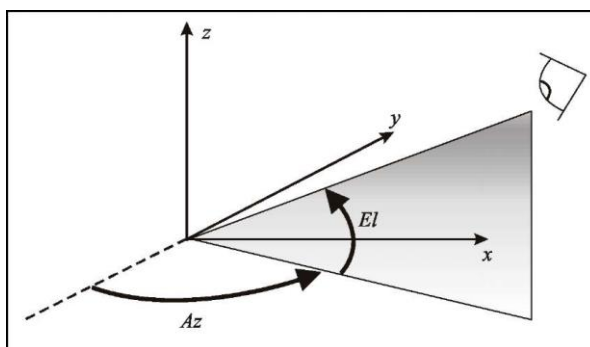


Рис. 4.16. Визначення точки зору

За замовчанням графіки поверхні зображуються для  $Az = -37,5^\circ$ ,  $Ei=30^\circ$ . Ці значення в командному вікні можна змінити функцією

```
view([Az Ei]).
```

Змінити точку зору зручніше інтерактивно мишею в графічному вікні. Режим просторового повороту вмикається пунктом **Rotate 3D** меню **Tools**, кнопкою **Rotate 3D** кнопкової панелі вікна. При переміщенні миші з натиснутою лівою кнопкою зображення буде синхронно повертатися. В лівому куті вікна при цьому динамічно відображаються значення кутів орієнтації точки зору.

Графічне вікно забезпечує ще одну потужну можливість налаштування вигляду просторових поверхонь – це засоби кнопкової панелі 3D перегляду. Зображення у вікні імітується роботою фотографічної камери. В панелі



можна керувати режимом відображення об'єкта. В *Matlab* використано два режими: ортографічний режим, в якому будується паралельна проекція, та перспективний режим, в якому будується перспективна проекція.

Функції 3D перегляду: *Orbit Camera* – обертанням камери навколо зображення. Дія є аналогом опції *Rotate 3D*, *Orbit Scene Light* – орієнтацією джерела світла відносно об'єкта, *Pan/Tilt Camera* – кутовий зсув об'єкта відносно камери, *Move Camera Horizontally/Vertically* – переміщення об'єкта в полі зору камери, *Move Camera Forward/Back* – зміна відстані між камерою та об'єктом, *Zoom Camera* – зміна збільшення камери, *Roll Camera* – поворот камери навколо своєї осі.

#### 4.4.2. Завдання для самостійного виконання

*Завдання 4.4.1.* Положення рухомого об'єкту визначається як

$$x = \left(\frac{t-15}{100} + 1\right) \sin(3t) \quad y = \left(\frac{t-15}{100} + 1\right) \cos(0.6t) \quad z = 0.3t^{1.6}$$

Зобразити траєкторію руху точки для  $0 < t < 30$ .

*Завдання 4.4.2.* Для функції  $z(x, y) = (\sin(y^2) + \cos(x^2))^{xy}$  намалювати в одному вікні в діапазоні аргументів від -1 до 1

- а) каркасну поверхню з невидимими лініями,
- б) каркасну поверхню без невидимих ліній,
- в) каркасну поверхню з ізолініями.

*Завдання 4.4.3.* Для функції  $z(x, y) = \frac{y^2}{3} - 1.5 \sin(2x)$  намалювати в одному вікні в діапазоні аргументів від -3 до 3

- а) поверхню-оболонку з ребрами,
- б) поверхню без ребер та плавним фарбуванням граней,
- в) поверхню з додатковим джерелом світла.

*Завдання 4.4.4.* Для функції  $z(x, y) = 3y + y^3 - x^3$  намалювати в одному вікні контурні графіки в діапазоні аргументів від -1 до 1

- а) графік з кількістю ізоліній 20,

- б) графік з фарбуванням,
- в) ізолінію  $5 = 3y + y^3 - x^3$ ,
- г) аксонометричний графік ізоліній.

## Практикум 4.5. Символьні обчислення

### 4.5.1. Засоби Matlab

Символьне перетворення *Matlab* проводиться засобами розширення *Symbolic Math Toolbox*, який призначений для проведення перетворення виразів, аналітичного розв'язання задач лінійної алгебри, диференціальних та інтегральних обчислень, отримання чисельного результату з підвищеною точністю. В *Matlab* використовуються можливості *CAS MuPAD*, яку розробники *Matlab* придбали та вбудували в *Matlab*. Запустити *MuPAD* можна командою **mupad**.

#### Символьні змінні та функції

Символьні змінні є об'єктами класу *sym object*.

Змінні та вирази повинні бути попередньо описані як символьні. Для визначення одної змінної чи виразу застосовується функція **sym('x', <'type'>)**. Аргументом є вираз/ім'я змінної, забраний в апострофи. Опціональний параметр **'type'** визначає тип змінної: **'real'** – дійсна змінна, **'positive'** – додатна дійсна змінна, **'unreal'** – комплексна змінна, **'clear'** – скидання типу до передумовленого. Розробник попереджає про майбутнє припинення підтримки функції **sym**.

Для визначення кількох змінних одночасно застосовується функція **syms x y, <type>**. Вирази, в яких використані символьні змінні, теж є символьними.

Наприклад, запис

```
>> x=sym('x','real'); syms a b;
```

```
>> c=sym(' (sin(x)+a)^2*(cos(x)+b)^2 ');
>> f = (sin(x)+a)^2*(cos(x)+ b)^2/sqrt(abs(a+b))
```

в першому рядку визначає символічну змінну «x» як дійсну та дві символічні змінні «a», «b» без визначення типу. В другому рядку визначено змінну «c» як символічний вираз. В третьому рядку визначено символічний вираз «f» простим поєднанням арифметичними діями символічних змінних «x», «a», «b».

Для отримання запису виразу в майже «природньому» вигляді призначена функція **pretty(ім'я виразу)**.

### Обчислення з символічними змінними

Символьні змінні можуть бути елементами матриць та векторів. В результаті створюються символічні матриці та вектори, до яких можна застосовувати матричні та поелементні операції.

```
» syms a b c d e f g h; A = [a b; c d]
```

```
» B = [e, f; g, h] ;
```

```
» C = A*B; C =  $\begin{bmatrix} a*e+b*g & a*f+b*h \\ c*e+d*g & c*f+d*h \end{bmatrix}$ 
```

```
» F = A.*B F =  $\begin{bmatrix} a*e & b*f \\ g*c & d*h \end{bmatrix}$ 
```

Символьний процесор дозволяє знаходити чисельне значення з підвищеною точністю. Для отримання чисельних значень символічних виразів призначена функція **vpa(S,D)**, де **S** – символічний вираз, **D** – опціональне значення кількості цифр у відповіді. За замовчанням виводиться 32 цифри.

Наприклад,

```
»c=sym('exp(1)');vpa(c)
```

```
2.71828182845904523536028747135
```

Параметр **D** функції **vpa** діє тільки для поточного виклику функції. Для глобального визначення кількості цифр слід використовувати функцію **digits(D)**. Виклик функції **digits** без аргументу повертає поточне значення **D**.

Більшість чисельних перетворень над матрицями можуть бути проведені в символьному вигляді також. Імена функцій символьних перетворень матриць співпадають з іменами функцій для чисельних дій: **det** – визначення детермінанта матриці, **inv** – знаходження зворотної матриці, **poly** – знаходження характеристичного полінома.

Наприклад,

```

» A = sym(' [a b c; d e f; g h j ] ');
» det(A)      j*a*e-a*f*h-j*d*b+d*c*h+g*b*f-g*c*e
» inv(A)
[(f*h-e*j)/(a*f*h-b*f*g-c*d*h+c*e*g-a*e*j+b*d*j), -(c*h-...
b*j)/(a*f*h-b*f*g-c*d*h+c*e*g-a*e*j+b*d*j), -(b*f-c*e)/...
(a*f*h-b*f*g-c*d*h+c*e*g-a*e*j + b*d*j)]
[-(f*g-d*j)/(a*f*h-b*f*g-c*d*h+c*e*g-a*e*j+b*d*j), (c*g-...
a*j)/(a*f*h-b*f*g-c*d*h+c*e*g-a*e*j+b*d*j), (a*f-c*d)/...
(a*f*h - b*f*g - c*d*h + c*e*g - a*e*j + b*d*j)]
[-(d*h-e*g)/(a*f*h-b*f*g-c*d*h+c*e*g-a*e*j+b*d*j), (a*h-...
b*g)/(a*f*h-b*f*g-c*d*h+c*e*g-a*e*j+b*d*j), -(a*e-b*d)...
/(a*f*h - b*f*g - c*d*h + c*e*g - a*e*j + b*d*j)]
» poly(A)
x^3+(-a-e-j)*x^2+(a*e-b*d-c*g-f*h+j*(a+e))*x+(a+e)*(c*g+...
f*h) - g*(a*c + b*f) - j*(a*e - b*d) - h*(c*d + e*f)

```

Символьні вирази обраховуються максимально точно, на відміну від чисельної математики. Кількісні значення символьного та чисельного розрахунків відображаються на екрані по різному. Символьні результати вирівнюються по лівому краю, чисельні – зображуються з відступом.

### Перетворення виразів

Складні математичні вирази можуть бути приведені до еквівалентних шляхом спрощення та перетворення.

Для групування символьних поліномів «р» по ступенях визначеної змінної призначена функція **collect(p, <'a'>)**. За замовчанням

змінною вважається «x». Для проведення дії з іншою змінною, її ім'я в апострофах слід поставити другим аргументом функції.

Наприклад,

»pretty( collect ( p , ' a ' ) )

$a^4 + 4xa^3 + (-1 + 6x^2)a^2 + (4x^3 + x)a + x^4 + (x-1)^3 - x^2 - 3 + x$

При виклику функції без опціонального параметра за замовчанням групує поліном за зниженням степеню «x» від 4 до 0. Виклик цієї функції з додатковим параметром «a» проводить групування та ранжування доданків відносно вказаної змінної «a».

Розкрити дужки та обраховувати значення символьних функцій в виразах можна за допомогою функції **expand (p)** :

» pretty ( expand ( p ) )

$x^4 + 4ax^3 + 6x^2a^2 + 4xa^3 + a^4 + x^3 - 4x^2 + 4x - 4 + ax - a^2$

Розкладається не тільки алгебраїчний поліном, а і тригонометричні, експоненційні, логарифмічні функції. Наприклад,

»pretty ( expand ( sym ( ' sin ( arccos ( 3 \* x ) ) + exp ( 2 \* log ( x ) ) ' ) ) )

$(1 - 9x^2)^{1/2} + x^2$

Для визначення множників символьного полінома «p», якщо множники мають раціональні коефіцієнти, слугує функція **factor(p)**:

»pretty ( factor ( sym ( ' x^5 + 13 \* x^4 + 215 / 4 \* x^3 + 275 / 4 \* x^2 - 27 / 2 \* x - 18 ' ) ) )

$1/4 (2x + 1) (2x - 1) (x + 6) (x + 4) (x + 3)$

Якщо вираз є натуральним числом, то функція проводить його розкладання на добуток простих чисел:

»factor ( sym ( ' 230010 ' ) ) ( 2 ) \* ( 3 ) \* ( 5 ) \* ( 11 ) \* ( 17 ) \* ( 41 )

Спрощення символьних виразів проводиться функцією **simplify (f, <n>, <Seconds, T>)**. Функція спрощує вирази, які вміщують тригонометричні, експоненціальні, логарифмічні, спеціальні: гіпергеометричну, Беселя, гамма-функцію. Опціональний параметр <n>

визначає максимальну кількість кроків спрощення. За замовчанням дорівнює 50. Другий опціональний параметр визначає максимальний час на дію функції.

Наприклад,

```
>> syms alpa beta
>> simplify(exp(c*log(sqrt(alpa+beta))))
ans=(alpa+beta)^(c/2)
```

Подивитись результати застосування різних способів спрощення виразу дозволяє функція **simple**. Виклик з пріврівнюванням ховає проміжні результати.

Наприклад,

```
» z=sym('cos(x)^2+sin(x)^2)^(1/2); simple(z^2)
simplify: 1
radsimp: cos(k)^2+sin(k)^2
combine(trig): 1
factor: cos(k)^2+sin(k)^2 expand: cos(k)^2+sin(k)^2
combine: 1
convert(exp):
(1/2*exp(i*k)+1/2/exp(i*k))^2-1/4*(exp(i*k)-
1/exp(i*k))^2
convert(sincos): cos(k)^2+sin(k)^2 convert(tan):
(1-tan(1/2*k)^2)^2/(1+tan(1/2*k)^2)^2+4*tan(1/2*k)^2/...
(1+tan(1/2*k)^2)^2
collect(k): cos(k)^2+sin(k)^2
ans = 1
>> c=simple(z^2)
c =1
```

Для спрощення громіздких виразів, розрахунку символічних виразів при визначених значеннях обраної змінної потрібно проводити підстановку в

вираз. Функція **subs (f, <OLD>, <NEW>)** проводить заміну в символьному виразі **f** змінної **<OLD>** на змінну **<NEW>**.

Наприклад, підстановка для отримання чисельного результату:

```
>> f = sym('exp(x^3 + 2*x^2 + x + 5)');
```

```
>> subs(f, 'x', 1.3) ans = 1.4392e+005
```

послідовна підстановка двох змінних:

```
>> f=sym('cos(a)+sin(b)');
```

```
>> f1=subs(f, 'a', 'x^2') f1 =cos(x^2) + sin(b)
```

```
>> subs(f1, 'b', pi/3) ans = cos(x^2) + 3^(1/2)/2
```

Представити функцію у вигляді розкладання в ряд Тейлора дозволяє функція **taylor(f, <n>, <a>, <x>)**. За замовчанням розраховується шість членів розкладання. Опціональний параметр **<n>** дозволяє визначити нову кількість. Без наявності третього параметра **<a>** проводиться розкладання в ряд Маклорена, наявність параметра призводить до розкладання в ряд Тейлора навколо точки **<a>**. Опціональний параметр **<x>** застосовується для функцій кількох змінних та визначає змінну, по якій проводиться розкладання.

```
>>syms x;taylor(exp(-x)) -x^5/120+x^4/24-x^3/6+x^2/2-x+1
```

```
>>taylor(exp(-x), 3) x^2/2 - x + 1
```

Знаходження символьних значень для сум виконується функцією **symsum(z, <k>, <kmin>, <kmax>)**. Добутків – **symprod(z, <k>, <kmin>, <kmax>)**. Перший параметр – символьний вираз доданка/множника, другий – ім'я індексу, третій та четвертий – нижня та верхня межі. Нескінченості визначаються вбудованими змінними **Inf**, **-Inf**.

Ряд функцій *Symbolic Math Toolbox* призначено для розв'язання задач диференціального та інтегрального обчислення. Границі функцій та послідовностей знаходить функція **limit(F, <x>, <a>, <'mode'>)**. За замовчанням функція шукає границю символьної послідовності **F(x)**, в точці  $x=0$ . Опціональний параметр **<x>** явно визначає змінну, опціональний

параметр **<a>** визначає точку, де шукати границю, опціональний параметр **'mode'** може мати значення **'left'** або **'right'**.

Наприклад,

```
» syms x
```

```
» limit((10 + x)^(1/x), x, 0)           ans = NaN
```

#### 4.6.2. Засоби MuPAD

СКА *MuPAD* інсталується разом з розширенням *Symbolic Math Toolbox Matlab* та може використовуватися автономно. Запустити *MuPAD* можна з командного вікна командою **mupad**.

Так, як в *Matlab* символічні перетворення, здійснюються ядром СКА *MuPAD*, відповідно, можливості СКА *MuPAD* є не меншими за ті, що реалізовані засобами розширення *Symbolic Math Toolbox*.

Символьні дії в можна проводити введенням команд в робоче поле пакета або за допомогою кнопок панелей. Використання кнопок панелей суттєво підвищує зручність проведення дій.

##### Оболонка пакета

Вікно *MuPAD* відкривається з назвою *Notebook* (рис. 4.17). Вигляд вікна є стандартним для *Windows* застосунків. Оболонка пакета містить розташоване згори вікно застосунку **меню, кнопоківі панелі Standard Bar, Format Bar, Command Bar, Find and Replace Bar**. В центрі оболонки розташовується *робоче командне вікно*. Внизу вікна розміщений *статусний рядок*. В *статусному рядку* відображається тип активної області робочого вікна та параметри введення.

Інформація в робочому вікні розміщується в текстовій *Text Paragraph* або командній області *Calculation*. Область займає один або кілька рядків. Кількість та порядок областей не обмежується.

В текстову область вводяться текстові пояснення, коментарі тощо. Візуальних позначок, які позначають область як текстову не передбачено. Абзаци тексту в текстовій області можуть бути відформатовані засобами кнопкової панелі **Format**.



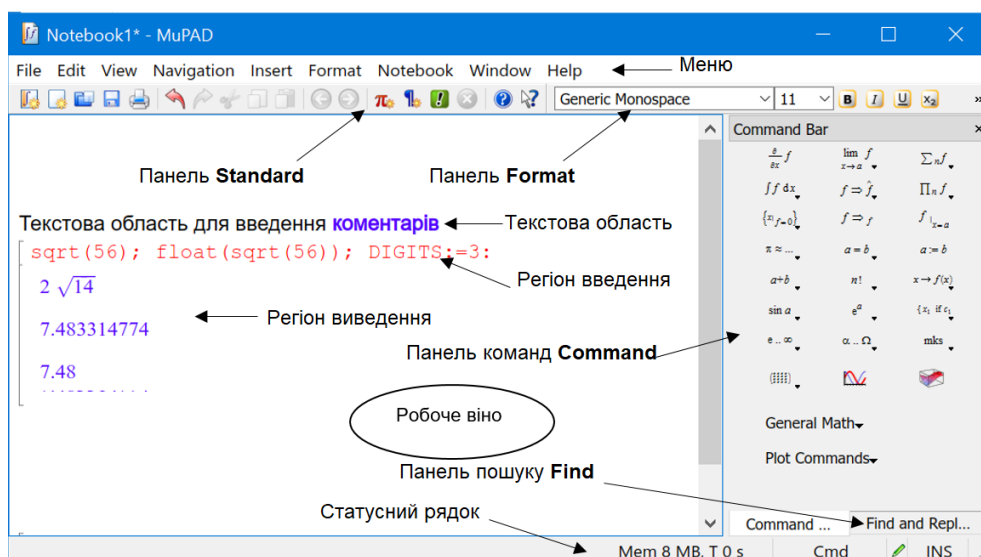




Рис. 4.19. Оболонка *MuPAD*

Для введення тексту достатньо встановити курсор за межами командної області та почати вводити текст.

Створити нову текстову область можна кнопкою  **Insert Text** панелі **Standard** або пунктом **Insert-Text Paragraph** меню.

Командна область є основною для введення команд і відображення результатів розрахунку. В ній користувачем вводяться команди з клавіатури або кнопкової панелі **Command** та відображаються обраховані результати. Візуально командна область позначається сірим символом квадратної дужки на початку рядка. Командна область розділена на два регіони. Згори знаходиться регіон введення команд, під ним знаходиться регіон виведення результатів. За замовчанням введені дані мають червоний колір, результати – синій. Формат шрифтів командної області може бути налаштований пунктом **Format-Characters** меню.


Створити нову командну область можна кнопкою  **Insert Calculation** панелі **Standard** або пунктом **Insert-Calculation** меню.

### Робота в пакеті

Знаком прирівнювання для надання значень слугує символ «:=». В одному рядку можуть записуватися кілька дій. Дії розділяються в рядку символом крапки з комою «;». Всі дії, які записані в рядку виводяться в

регіон виведення. Для того, щоб не виводити дію в регіон виведення, слід в кінці опису дії поставити символ двокрапки «:».

Натискання клавіші в кінці рядка виводить результат дій рядка в регіон виведення.

Зміна даних в одній командній області враховується автоматично в інших командних областях. Обрахувати командні області можна кнопкою  **Evaluate** панелі або пунктами **Notebook-Evaluate From Beginning/Evaluate To End/ Evaluate All** меню.

Введені змінні та вирази витираються з пам'яті командою **delete**. В якості аргументів через кому перераховуються визначені імена.

Для введення матриць призначена функція **matrix**. Рядки матриці забираються в квадратні дужки. Між собою рядки розділяються комами.

Наприклад,

```
A := matrix([[1, 2], [3, 4], [5, 6]]);
B := matrix([[1, 2, 3], [4, 5, 6]])
```

$$\begin{pmatrix} 1 & 2 \\ 3 & 4 \\ 5 & 6 \end{pmatrix}$$
$$\begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix}$$

Отримати довідку про використання функцій пакета можна функціями **info(ім'я функції)** та **?ім'я функції**. Функція **info** виводить коротке пояснення в регіон виведення. Функція **?...** відкриває відповідну сторінку *html help* пакета. Утримання на імені функції в регіоні введення виводить спливаючу підказку.

```
info(sqrt)          sqrt(56)
sqrt -- the square 2  sqrt -- the square root function
                    sqrt(z)
```

За замовчання пакет виводить результат в компактному вигляді. За потреби отримати чисельне значення слід використати функцію **float**. Кількість знаків чисельного значення визначається системною змінною **DIGITS**. За замовчанням **DIGITS=10**.

Наприклад,

```
sqrt(56); float(sqrt(56)); DIGITS:=3: float(sqrt(56))  
2  $\sqrt{14}$   
7.483314774  
7.48
```

За замовчанням ознакою уявної одиниці в регіоні введення є символ прописної літери і «I».

Назва більшості функцій символічних перетворень в пакеті не відрізняється від синтаксису відповідних команд в СКА *Matlab*. Для більшості функцій синтаксис аргументів теж є аналогічним.

Найзручніше проводити символічні перетворення введенням назв функцій з панелі **Command**. На панелі дії **General Math** згруповані в логічні групи (Рис. 4.20) **Expand**, **Factor**, **Normalize**, **Evaluate**, **Simplify**, **Rewrite**, **Combine**, **Calculus**, **Matrices**, **Polynomials**. Кнопки найбільш часто вживаних дій виведені безпосередньо на панель.

Натискання кнопки на панелі **Command** призводить до появи в регіоні введення шаблону відповідної функції. В шаблоні аргументи позначені символом дієзу «#». Для проведення дії в поля аргументів користувачем повинні бути вписані фактичні дані. Перемикання між аргументами можна проводити мишею або клавішею *Tab*.

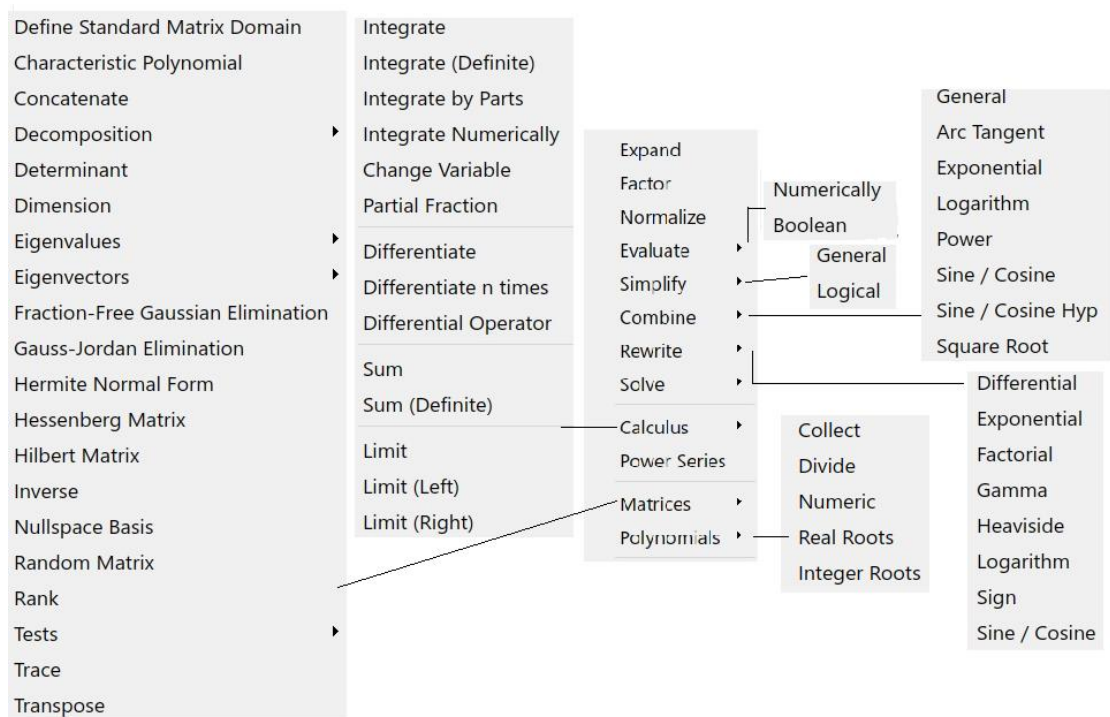


Рис. 4.20. Склад панелі **Command Bar**

Матричні перетворення не потребують додаткових описів.

Наприклад, для квадратних матриць  $A$ ,  $B$  добуток матриць, визначник матриці  $A$ , обернена матриця  $A$ , характеристичний поліном матриці  $B$  будуть мати наступний вигляд:

```
A := matrix([[a, b], [c, d]]):
B := matrix([[e, f], [g, k]]): A*B;det(A);1/A;
linalg::charpoly(A, x)
```

$$\begin{pmatrix} a e + b g & a f + b k \\ c e + d g & c f + d k \end{pmatrix}$$

$$a d - b c$$

$$\begin{pmatrix} \frac{d}{a d - b c} & -\frac{b}{a d - b c} \\ -\frac{c}{a d - b c} & \frac{a}{a d - b c} \end{pmatrix}$$

$$x^2 + (-a - d)x + a d - b c$$

Функція **collect** проводить групування символічних поліномів по степенях.

Функція **expand** розкриває дужки в символічних виразах .

Функція **factor** визначає множники символічного полінома.

Функції **simplify** та **Simplify** спрощують символічні вирази. Функція **simplify** працює швидше. Функція **Simplify** дозволяє спрощувати більш складні вирази та має опціональні аргументи. Аргумент визначає **Steps** кількість кроків спрощення. Аргумент **All** дозволяє отримати інший варіант запису результату.

```
f:=exp((cos(x)^2 - sin(x)^2)*sin(2*x)*(exp(2*x)-2*exp(x) + 1)/(exp(2*x) - 1)):
simplify(f);Simplify(f);Simplify(f,Steps=250)
```

$$e^{\frac{\sin(4x)(e^x-1)}{2(e^x+1)}}$$

$$e^{-\frac{\sin(4x)-\sin(4x)e^x}{2e^x+2}}$$

$$e^{\frac{\sin(4x)(e^x-1)}{2(e^x+1)}}$$

Функція **subs (p,old=new)** проводить заміну в символічному виразі **p** змінної **old** значенням. Отримати обчислений результат перетворення можна застосуванням опціонального аргументу **EvalChanges** або викликом після функції обчислення **eval (%)**.

```
f:=sin(x^2) + 1:subs(f, x = 0): subs(f,x=0);eval(%);subs(f,x=0, EvalChanges)
```

$$\sin(0) + 1$$

$$1$$

$$1$$

Функція **series (p,x=a,n)** розкладає символічний вираз **p** в степеневий ряд Тейлора.

Функція **sum** розраховує символічне значення суми.

Функція **product** символічне значення добутку.

Функція **limit** шукає границю символічного виразу.

Функція **diff** обчислює першу похідну символічного виразу. Диференціювання більших порядків проводиться послідовним викликом функції або повторенням через кому аргументів функції.

Функція **int** обчислює інтеграл від символічного виразу.

### 4.5.3. Завдання для самостійного виконання

Завдання 4.5.1. Спростити вираз в *Matlab* та в *MuPAD*.

- а)  $\log(x)$ -х  $\log(x)$ -х групуванням відносно  $\log(x)$  ,  
 б)  $xy + axy + ux^2 - aux^2 + ax$  групуванням відносно  $x$  .

Завдання 4.5.2. Розкрити вираз в *Matlab* та в *MuPAD*.

- а)  $\sin(x + y)$  б)  $\cos(3 \arccos(x))$  .

Завдання 4.5.3. Розкласти вираз на множники в *Matlab* та в *MuPAD*.

- а)  $x^3 + 2$  б)  $x^4 - y^4$ .

Завдання 4.5.4. Спростити вираз в *Matlab* та в *MuPAD*.

- а)  $(1-x^2) / (1-x)$  б)  $\cos(x)^2 + \sin(x)^2$  в)  $\cos(2x) - \sin(2x)$ .

Завдання 4.5.5. Отримати символічний результат та чисельне значення при  $x=3$  в *Matlab* та в *MuPAD*.

- а)  $A \cdot B$  , б)  $A/B$  , в)  $A+B$ , де

$$A = x(x^2 + 6x + 10) + 6, \quad B = (x-3)^2 + 10x - 5.$$

Завдання 4.5.6. Розрахувати в *Matlab* та в *MuPAD*.

- а)  $e^\pi$  та  $\pi^e$ , з точністю до 25 знака. Яке значення більше?  
 б)  $2709/1024$ ,  $10583/4000$ ,  $2024/765$ . Який результат ближчий до 1.7?

Завдання 4.5.7. Розрахувати значення сум в *Matlab* та в *MuPAD*.

- а)  $\sum_1^{10} \frac{1}{k^4}$  , б)  $\sum_1^{\infty} \frac{1}{k^4}$

Завдання 4.5.8. Розкласти вираз в ряд Тейлора навколо точки  $c$  з  $n$  доданками в *Matlab* та в *MuPAD*.

- а)  $e^x$ ,  $n=4$ ,  $c=0$       б)  $\operatorname{tg}(x)$ ,  $n=6$ ,  $c=0$       в)  $\ln(x)$ ,  $n=4$ ,  $c=1$

Завдання 4.5.9. Знайти похідні функцій в *Matlab* та в *MuPAD*.

- а)  $\frac{2x-1}{x^2+1}$  , б)  $\sqrt{1+x^4}$  , в)  $\operatorname{arctg}(x^2+1)$

Завдання 4.5.10. Визначити границю в *Matlab* та в *MuPAD*.

- а)  $\lim_{x \rightarrow \infty} \frac{1-e^{-x}}{x}$  , б)  $\lim_{x \rightarrow 1} \frac{1-x}{\ln(x)}$  , в)  $\lim_{x \rightarrow 1} (1-x)^{\frac{1}{1-x}}$  , г)  $\lim_{x \rightarrow 1} \left(\frac{1}{1-x}\right)$

## Практикум 4.6. Експорт-імпорт даних

### 4.6.1. Теоретичні положення

#### Низькорівневий файловий обмін.

Визначення специфікацій файлів засобами віконного діалогу проводиться функціями **uigetfile**, **uiputfile**

Функція

```
[FileName, Path<, Filter>]=  
uigetfile<(FilterSpec, Title, Default)>
```

виводить модальне системне діалогове вікно для відкриття файлу.

Функція

```
[FileName, Path<, Filter>]=  
uiputfile<(FilterSpec, Title, Default)>
```

виводить модальне системне діалогове вікно для введення імені файлу для збереження. Якщо значенням аргументу є ім'я файлу, то це ім'я буде вказано за замовчанням в полі вибору файлу вікна.

Відкриття файлу проводиться функцією **fopen**.

```
<[>fileID<, errmsg]>=  
fopen(filespec<, permission, fmt, encoding>)
```

відкриває доступ до файлу, який визначений в обов'язковому аргументі **filespec**, який є масивом символів та містить повну специфікацію файлу. Опціональний аргумент **permission** є масивом символів та визначає режим, в якому може бути використаний файл. Файл може бути відкритий для зчитування **'r'**, для запису **'w'**, для додавання інформації в кінець існуючого файлу **'a'**, для зчитування та запису **'r+'**, **'w+'**, для зчитування та додавання інформації **'a+'**. В разі відсутності аргументу, вважається заданим режим зчитування **'r'**.

Опціональний аргумент **encoding** є масивом символів, який визначає кодову сторінку текстових даних. Підтримує більшість з доступних в ОС кодових сторінок. Наприклад, **'UTF-8'**, **'ISO-8859-1'**, **'windows-**

1251', 'GBK', 'IBM866', 'KOI8-U', 'Macintosh' тощо. В разі відсутності використовується поточна кодова сторінка системи.

Опціональний аргумент **fmt** є масивом символів та визначає порядок обробки байтів в словах для бінарної інформації. В разі відсутності аргументу застосовується 'n', 'native' поточна обробка системи.

Опціональний результат **errmsg** містить системне повідомлення про помилку. В разі успішної операції є порожнім рядком.

Зчитування інформації проводиться функціями **fgetl**, **fgets**, **textscan**.

Функція **line = fgetl(fileID)** зчитує рядок з текстового файлу. З рядка прибираються службові символи переведення рядка. В разі досягнення кінця файлу повертається чисельне значення -1.

Функція **line = fgets(fileID <,nchar>)** зчитує рядок з текстового файлу. З рядка НЕ прибираються службові символи переведення рядка. В разі досягнення кінця файлу повертається чисельне значення -1. Опціональний аргумент **nchar** визначає кількість символів в рядку, які будуть зчитуватися.

Функція

```
<[> C <,position]>=textscan(fileID,format<,N, Name,Value>)
```

зчитує дані з текстового файлу в масив комірок. Файл визначається ідентифікатором файлу **fileID**. Формат даних для зчитування визначається аргументом у вигляді масиву символів **format**. Синтаксис опису формату є аналогічним функціям **sprintf**, **fprintf**.

Функція проводить зчитування доки дані відповідають вказаному в аргументі **format** формату.

Опціональний аргумент **N** є додатним цілим числом, яке визначає кількість разів зчитування визначеного формату. Для продовження зчитування слід повторити виклик функції з тим же ідентифікатором файлу.

Формат даних кодується записом у вигляді **%\*L.MT**



В описі формату зчитування припустимі дані таких типів **T**, як цілого типу **%d**, цілого типу без знака **%u**, дійсного типу одинарної точності **%f**, дійсного типу подвійної точності **%n**, одиночні символи **%c**, текст у вигляді масиву символів **%s**, дати **%D** тощо.

Поле **L** визначає загальну кількість символів в даному, поле **M** – кількість після коми.

Комплексні числа записуються для зчитування у формі **±<real>±<imag>i/j**.

Запис **%\*k**, де **k** – ідентифікатор типу даних, показує, що описаний код НЕ буде зчитуватися.

Запис для не числових типів даних **%\*ns** показує, що в даному НЕ буде зчитуватися **n** символів.

Опціональні пари аргументів **Name, Value** визначають додаткові налаштування зчитування в термінах об'єктного програмування "властивість-значення".

Властивість **'CollectOutput'** визначає, що всі дані рядка повинні збиратися в єдиний масив символів. Припустимими значеннями є **true** або **false**.

Властивість **'CommentStyle'** визначає символи, між якими дані вважаються коментарем та НЕ зчитуються. Наприклад, **'CommentStyle', {'/\*', '\*/'}, 'CommentStyle', '%'**.

Властивість **'Delimiter'** визначає символ розділення даних в рядку. Без вказання символу розділення таким вважається символ пробілу.

Наприклад, **'Delimiter', {';', '\*'}, 'Delimiter', ', ', 'Delimiter', '\t'**.

Властивість **'EmptyValue'** визначає яким числом замість **NaN** буде замінюватися порожнє поле в рядку

Властивість **'EndOfLine'** визначає яким символом буде позначатися кінець рядка замість стандартних **'\n', '\r', '\r\n'**.

Властивість **'HeaderLines'** визначає кількість рядків з початку файлу, які не будуть зчитуватися.

Властивість **'TreatAsEmpty'** визначає для чисельних значень які з них будуть замінені на **NaN**. Значення записуються як масив комірок або вектор-рядок символів.

Властивість **TextType** визначає для текстових даних значенням **'char'** (за замовчанням) їх збереження у вигляді масиву символів, значенням **'string'** – збереження у вигляді масиву комірок.

Результатом функції є масив комірок **C**. Опціональний результат містить номер останнього зчитаного рядка

Наприклад, застосування функції до текстового файлу з ідентифікатором **id**, в якому міститься рядок **1.0 2.0 3.0 4.0**

```
C = textscan(id, '%f');
```

результатом дає масив комірок розміром 1x1. Елемент масиву є вектором-стовпцем **[1;2;3;4]**.

Скидання покажчика зчитування на початок файлу проводиться функцією **frewind(fileID)**.

Закриття файлу проводиться функцією

```
<s> = fclose(fileID <,all>).
```

Функція закриває файл з внутрішнім ідентифікатором **fileID** та вивільняє змінну **fileID**. Опціональний аргумент **'all'** вказується замість ідентифікатора файлу та викликає закриття всіх відкритих файлів. Опціональний результат **s** має значення 0, якщо операція пройшла успішно та -1 в інших випадках.

### **Програмні засоби високого рівня файлового обміну.**

Програмні засоби у вигляді вбудованих функцій обміну високого рівня дозволяють скоротити обсяг програмування для організації обміну стандартизованою інформацією. У вбудованих функціях обміну високого рівня дії відкриття файлу, зчитування, перетворення даних, закриття файлу містяться всередині функцій.

## Обмін з текстовими файлами

Функція

```
M = csvread (filespec <,R1,C1,[R1 C1 R2 C2]>)
```

зчитує в масив **M** з текстового файлу дані у вигляді таблиці чисельних даних з розділенням комою. Обов'язковим аргументом функції є повна специфікація файлу **filespec** у вигляді масиву символів. В разі відсутності опціональних аргументів проводиться зчитування всього файлу.

Опціональні аргументи **R1**, **C1**, **R2**, **C2** задають діапазон зчитування. **R1 (C1)** визначає номер рядка (стовпця) початку зчитування, **R2 (C2)** – номер рядка (стовпця) кінця зчитування. Числами нумерація проводиться з нуля. Для зчитування з рядка **R1** стовпця **C1** до кінця файлу другий опціональний аргумент-вектор не записується. Для зчитування частини даних всередині файлу записуються обидва опціональні аргументи.

Наприклад, застосування функції до текстового файлу **test.dat**, в якому містяться наступні чисельні дані:

```
1.1, 2.0, 3.3
```

```
4.0, 5.5, 6.5
```

```
7, 8.8, 9.9
```

```
M = csvread('test.dat')
```

результатом дає матрицю **M** розміром 3x3 з дійсними елементами подвійної точності:

```
M =
```

```
1.0999999999999994 2.0000000000000000 3.2999999999999997
```

```
4.0000000000000000 5.5000000000000000 6.4800000000000000
```

```
7.0000000000000000 8.7999999999999994 9.9000000000000000
```

```
Оператор csvwrite (filespec, M <,R1,C1>)
```

записує в текстовий файл з розділенням комою чисельні дані з масиву **M**. Обов'язковими аргументами функції є повна специфікація файлу **filespec**

у вигляді масиву символів та ім'я масиву **M**. Опціональні аргументи **R1**, **C1** задають числом номер рядка (стовпця) початку запису. Числами нумерація проводиться з нуля. В результуючій матриці на місцях до **R1**, **C1** записуються коми.

Запис `csvwrite('scv.dat',M)` результатом дає текстовий файл `scv.dat`

```
81.1,52,73.3
```

```
36,67.5, 6.48
```

```
58.5,93.6, 2.7
```

Функція

```
M = dlmread (filespec, <del,R1,C1>)
```

зчитує в масив **M** з текстового файлу дані у вигляді таблиці чисельних даних з розділенням. Обов'язковим аргументом функції є повна специфікація файлу **filespec** у вигляді масиву символів. В разі відсутності опціональних аргументів проводиться зчитування всього файлу.

Аргумент **del** визначає символ, який слугує розділювачем даних в файлі. Для знака табуляції застосовують символ `'\t'`. Без вказання аргументу функція автоматично визначає розділювач.

Опціональний аргумент **R1,C1** визначає індекси комірки, з якої треба зчитувати дані. Перед аргументом слід визначити символ розділювача. В разі визначення аргументу вектором [**R1 C1 R2 C2**] зчитування проводиться в діапазоні рядків від **R1** до **R2** та стовпців – від **C1** до **C2**. Замість вектора можна визначати діапазон символами за правилами адресації в *MS Excel* 'літера стовпця номер рядка'. Комірка 0,0 має адресу **A1**.

Оператор

```
dlmwrite (filespec,M<,-'append'><,del,R1,C1,Name,Value >)
```

записує в текстовий файл з розділенням чисельні дані з масиву **M**. Обов'язковими аргументами функції є повна специфікація файлу **filespec** у вигляді масиву символів та ім'я масиву **M**. Опціональний аргумент **del** визначає символ розділення. В разі відсутності аргументу **del** вважається, що для розділення використовується кома. Для знака табуляції застосовують

символ `'\t'`. Опціональний аргумент `'append'` дозволяє додавати дані в файл. Опціональний аргумент `R1, C1` визначає стартову комірку, з якої в файл будуть вставлені дані з матриці **M**. Перед стартовою коміркою додаються знаки розділення.

Застосування засобів об'єктного програмування «властивість-значення» дозволяє провести тонке налаштування форматів запису даних в файл. Можуть використовуватися властивості `'delimiter'` – аналог аргументу **del**, `'roffset'` – в аналог аргументу **R1**, `'coffset'` – аналог аргументу **C1**, `'precision'` – формат запису чисел. Формат визначається або числом-кількістю знаків після коми, або рядком `'%10.5f'` аналогічно функціям **sprint**, **fprintf**.

Функція **T = readtable(filespec<,Name,Value>)**

зчитує в таблицю **T** з текстового файлу таблично організовані дані. Обов'язковим аргументом функції є повна специфікація файлу **filespec** у вигляді масиву символів. Функція може застосовуватися для файлів *MS Excel*. Обов'язковим аргументом функції є повна специфікація файлу **filespec** у вигляді масиву символів.

Функція є надбудовою над функцією **textscan**.

Опціональні пари аргументів **Name, Value** визначають додаткові налаштування зчитування в термінах об'єктного програмування «властивість-значення».

Властивості `'CollectOutput'`, `'CommentStyle'`, `'Delimiter'` (в разі відсутності функція намагатиметься визначити символ автоматично), `'EmptyValue'`, `'EndOfLine'`, `'HeaderLines'` (для текстових файлів), `'TreatAsEmpty'`, `'TextType'`, `'Format'` є аналогічними функції **textscan**.

Властивість `'ReadVariableNames'` визначає значенням **true** зчитування першого рядка у вигляді імен змінних таблиці та значенням зчитування всіх стовпців з другого елементу в них. В разі відсутності функція призначає змінним стовпців таблиці імена за замовчанням на зразок `'var1'`.

Властивість **'ReadRowNames'** визначає значенням **true** зчитування першого стовпця у вигляді імен рядків та значенням **false** (за замовчанням) зчитування всіх рядків без виділення першого елементу з них.

У випадку, коли обидві властивості **'ReadVariableNames'** та **'ReadRowNames'** є **true**, функція зберігає елемент файлової таблиці як першу назву властивості таблиці **\*.Properties.DimensionNames**.

Властивість **'FileType'** визначає для випадку, коли в специфікації файлу не вказано або відмінне від **.txt**, **.dat**, **.csv** розширення, значенням **'text'** текстові файли та значенням **'spreadsheet'** файли, розширення яких є відмінним від **.xls**, **.xlsb**, **.xlsm**, **.xlsx**, **.xltm**, **.xltx**, **.ods**.

Властивість **Encoding** визначає кодову сторінку для зчитаної текстової інформації з текстових файлів. Формат є аналогічним функції **fopen**.

Наприклад, для текстового файла **dat.txt**, який містить таблицю з першим рядком у вигляді назв стовпців

```
Level L H V Corr
Level1 1.2 3.4 Inf Yes
Level2 5.4 3.2 1e3 No
Level3 4.2 21 0.1 Yes
```

запис

```
T=readtable('e:\dat.txt','ReadVariableNames',false)
```

зчитає в таблицю **T** розміром 4x5. Змінні стовпців набудуть імена за замовчанням. У всіх комірках таблиці будуть текстові рядки.

```
T = 4x7 table
Var1      Var2  Var3  Var4  Var5
-----
'Level'   'L'    'H'   'V'   'Corr'
'Level1'  '1.2'  '3.4' 'Inf' 'Yes'
'Level2'  '5.4'  '3.2' '1e3' 'No'
'Level3'  '4.2'  '21'  '0.1' 'Yes'
```

Оператор **writetable(T <,filespec, Name,Value>)** записує в текстовий файл таблицю **T**. Обов'язковим аргументом функції є ім'я таблиці **T**. В такому разі створюється файл з назвою **T.txt** з розділенням комою. Опціональний параметр у вигляді масиву символів **filespec** є повною специфікацією файлу. Розширення файлу може бути **.txt, .dat, .csv** для текстових файлів або **.xls, .xlsm, .xlsx** – для файлів *MS Excel*. Для інших розширень слід застосовувати уточнення властивістю **'FileType'**.

Опціональні пари аргументів **Name,Value** застосовуються для форматування даних у файлі. Значення властивостей є аналогічним функції **readtable**. Припустимими є наступні властивості: **'FileType', 'WriteVariableNames', 'WriteRowNames', 'DateLocale', 'Encoding', 'Delimiter'**.

Властивість **'QuoteStrings'** значенням **true** замінює масиви символів в апострофах в файлі на рядки в подвійних лапках. Значення за замовчанням **false** записує текстову інформацію в файл без апострофів та лапок.

### Обмін з файлами MS Excel

Функція **readtable** в режимі **spreadsheet** зчитує дані з *MS Excel* файлів. Слід взяти до уваги, що СКМ не може мати імена змінних з кириличними символами, тому назви стовпців таблиці потрібно давати латиницею.

Властивість **'Sheet'** визначає аркуш *MS Excel*, з якого буде проводитися зчитування. Значення може задаватися натуральним числом-номером аркуша або масивом символів з його ім'ям.

Властивість **'Range'** визначає масивом символів діапазон даних для зчитування. Визначається в термінах адресації *MS Excel*. Одна адреса **'Corner1'** означає, що зчитування буде проводитися з неї до кінця файлу. Дві адреси **'Corner1:Corner2'** визначають прямокутну область з початком **'Corner1'** та закінченням **'Corner2'**.

Функція **writetable** в режимі **spreadsheet** записує дані з таблиці в файл *MS Excel*. За замовчанням запис проводиться з комірки **A1** першого аркуша таблиці *MS Excel*.

Функція

```
<[>num <,txt,row]>=  
xlsread(filename <,sheet,xlRange,'basic', -1>)
```

зчитує в масив **num** дані з файлу *MS Excel*. Масив символів з повною специфікацією файлу **filename** є обов'язковим аргументом. Опціональний аргумент **sheet** є масивом символів або натуральним числом. Він визначає аркуш книги *MS Excel*, з якого треба проводити зчитування. Опціональний аргумент **xlRange** є масивом символів, який визначає діапазон комірок *MS Excel*, з яких буде проводитися зчитування. Наприклад, **'D2:H4'**.

Опціональний аргумент **basic** застосовується, коли на комп'ютері не встановлено *MS Excel*. Не визначені попередні аргументи записуються апострофами.

Опціональний аргумент **-1** використовується для того, щоб користувач обрав дані для зчитування інтерактивно в *MS Excel*. Не може застосовуватися разом з іншими опціональними аргументами. Наприклад,  
**num = xlsread(filename,-1)**.

Опціональний результат **txt** є масивом комірок з текстовими даними з *MS Excel*.

Опціональний результат **raw** є масивом комірок з даними з *MS Excel*. Отримання опціональних результатів значно збільшує час виконання операції. Рекомендується за потреби отримання результатів у вигляді масивів комірок застосовувати режим **basic**.

Функція

```
<[status,message]>=  
xlswrite (filename ,A <,sheet,xlRange>)
```

записує дані з масиву **A** в файл *MS Excel*, який визначено масивом символів з повною специфікацією файлу **filename**. Без опціональних аргументів запис проводиться в першу комірку першого аркуша книги *MS Excel*.




Опціональний аргумент **sheet** є масивом символів або натуральним числом. Він визначає аркуш книги *MS Excel*, в який треба проводити запис. Опціональний аргумент **xlRange** є масивом символів, який визначає діапазон комірок *MS Excel*, з яких буде проводитися зчитування. Наприклад, 'D2:H4'. Має обов'язково містити символ двокрапки ':'.

Опціональний результат **status** є цілим числом, яке визначає статус операції. Значення 1 показує, що операція запису була успішною, значення 0 показує помилку запису.

Опціональний результат **message** є структурою, яка містить системне повідомлення про результат операції.

### Засоби інтерактивного файлового обміну

Для інтерактивного імпорту даних з файлів використовується додаток *Import Tool*. Додаток викликається командою **uiimport** з командного вікна, кнопкою  стрічки *HOME*.

Покажемо функціонування додатку на прикладі зчитування текстового файлу '**dat.txt**', який використовувався в наведених вище прикладах. Файл містить як числові, так і текстові дані. Розділювачем слугує символ коми, але в рядках між елементами є пробіли.

Додаток розпізнає тип даних автоматично, про що свідчить активований перемикач *Delimited* на стрічці (рис. 4.21).

В центрі вікна показана таблиця даних. Адресація комірок таблиці проводиться в стилі *MS Excel*. Імена можна змінити. Режим редагування імен вмикається подвійним щигликом на відповідному імені.

Діапазон даних, які планується імпортувати, виділено рамкою та відображено в полі *Range* стрічки. Змінити діапазон можна зміною адрес в полі стрічки або перетягуванням рамки.

Дані з файлу можуть бути зчитані в матрицю *Numeric Matrix*, таблицю *Table*, масив комірок *Cell Array*, текстовий масив *String Array*, у вектори стовпців *Column vectors*. Тип результату визначається пунктом випадаючого списку *Output Type* стрічки (рис. 4.22 в).

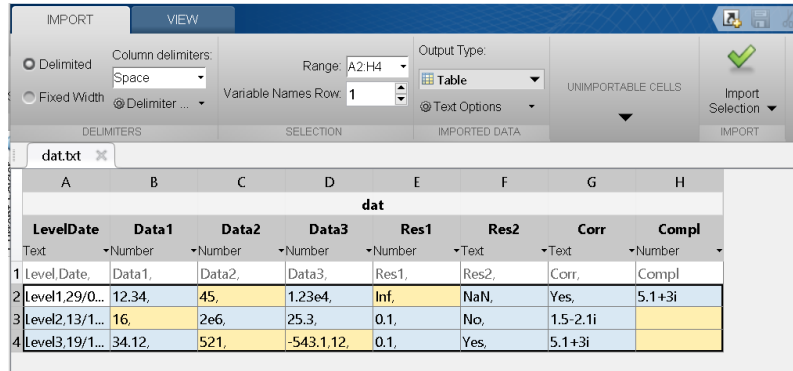


Рис. 4.21. Інтерактивне зчитування файлу

Під іменами змінних стовпців виведений тип даних для елементів відповідних стовпців. Тип даних можна змінити у вікні типів даних (рис. 4.22 а). Вікно відкривається натискання позначки трикутника ліворуч від вказаного для стовпця типу даних.

Дані з файлу можуть бути зчитані в матрицю *Numeric Matrix*, таблицю *Table*, масив комірок *Cell Array*, текстовий масив *String Array*, у вектори стовпців *Column vectors*. Тип результату визначається пунктом випадаючого списку *Output Type* стрічки (рис. 4.22 в).

Символ розділення можна змінити на потрібний в випадаючому списку *Column Delimiter* (рис. 4.22 б).

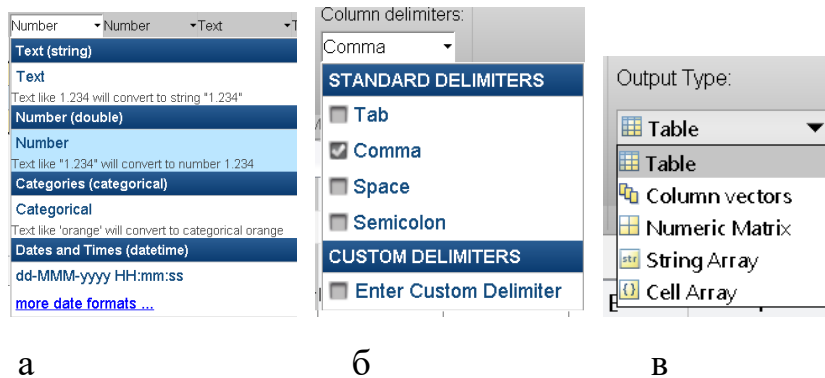


Рис. 4.22. Вікна керування інтерактивним імпортом: а- визначення типів даних; б – визначення розділення; в – тип результату

Елементи закладки *UNIMPORTABLE CELLS* стрічки керують діями з нерозпізнаними даними (рис. 4.23). В списку задаються символи, на які будуть замінені *Replace* нерозпізнані та відсутні дані *unimportble cells* та дії

виключення рядків *Exclude rows with* та/або стовпців *Exclude columns with* з відсутніми даними *Blank Cells* або нерозпізнаними даними *unimportable cells*.

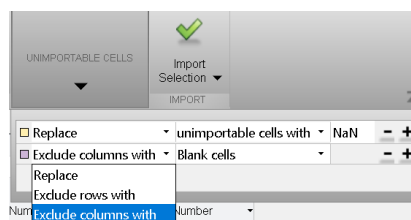


Рис. 4.23. Дії з нерозпізнаними даними

#### 4.6.2. Завдання для самостійного виконання

*Завдання 4.6.1.* Написати скрипт-файл для зчитування в масив  $M$  даних з файлу завдання 3.7.1 функцією високого рівня. Створити матрицю копіюванням горизонтально. Записати новоутворену матрицю в текстовий файл функцією високого рівня.

*Завдання 4.6.2.* Написати скрипт-файл зчитування в масив  $M$  даних з файлу завдання 3.7.2 функцією високого рівня. Створити матрицю копіюванням вертикально. Записати новоутворену матрицю в текстовий файл функцією високого рівня.

*Завдання 4.6.3.* Написати скрипт-файл зчитування в таблицю з назвами рядків та стовпців даних з файлу вступу до КПІ завдання 1.1.3 без шапки та першого рядка років функцією високого рівня. Розрахувати конкурс.

*Завдання 4.6.4.* Створити матрицю  $4 \times 4$   $M_{i,j} = \frac{i+j}{3}$ . Записати її;

а) в один текстовий файл з розділенням пробілом з шириною колонки 6 символів, шириною мантиси 3 символи функцією низького рівня, в другий – функцією високого рівня.

б) в один текстовий файл з розділенням комою з шириною колонки 8 символів, шириною мантиси 2 символи функцією низького рівня, в другий – функцією високого рівня.

*Завдання 4.6.5.* Написати скрипт-файл зчитування даних з файлу завдання 4.6.4. функцією низького рівня. Ім'я файлу вводити стандартним віконним діалогом.

*Завдання 4.6.6.* Зчитати дані з файлу *MS Excel* завдання 2.2.1 в вектори  $x$ ,  $y$ . Побудувати графік

*Завдання 4.6.7.* Зчитати дані першого аркуша «total\_list» завдання 2.3.1 . Розрахувати статистику аналогічно аркушу 3 «total» завдання 2.3.1.

## Література

1. Кравченко, І. В. Інформаційні технології. Системи комп'ютерної математики [Електронний ресурс]: навчальний посібник для студентів спеціальності «Автоматизація та комп'ютерно-інтегровані технології» / І. В. Кравченко, В. І. Микитенко; КПІ ім. Ігоря Сікорського. – Електронні текстові дані (1 файл: 5,44 Мбайт). – Київ : КПІ ім. Ігоря Сікорського, 2018. – 243 с. – Назва з екрана. Режим доступу <https://ela.kpi.ua/handle/123456789/22913>.
2. Microsoft Official Academic Course. MICROSOFT WORD 2013. MOS EXAM 77-418. - Wiley. - 2014. – 524 p.
3. УКРАЇНСЬКИЙ ПРАВОПИС. [Електронний ресурс]. ЗАТВЕРДЖЕНО Українською національною комісією з питань правопису (протокол № 5 від 22 жовтня 2018 р.) - 2019. – 282 с. Назва з екрана. - Режим доступу [https://mon.gov.ua/storage/app/media/zagalna\\_serednya\\_202019.pdf](https://mon.gov.ua/storage/app/media/zagalna_serednya_202019.pdf)
4. ДСТУ ГОСТ 2.104:2006 Єдина система конструкторської документації. Основні написи. Чинний від 2007-07-01. - Київ: ДП "УкрНДНЦ".- 2006. - 23с.
5. ДСТУ 3008:2015 Інформація та документація. ЗВІТИ У СФЕРІ НАУКИ І ТЕХНІКИ. Структура та правила оформлювання. Чинний від 2017-07-01. - Київ: ДП "УкрНДНЦ".- 2016. - 31с.
6. ДСТУ 8302:2015 Інформація та документація. БІБЛІОГРАФІЧНЕ ПОСИЛАННЯ. Загальні положення та правила складання. Чинний від 2016-07-01. - Київ: ДП "УкрНДНЦ". - 2016. - 20с.
7. Microsoft Official Academic Course. MICROSOFT EXCEL 2013. MOS EXAM 77-420. - Wiley. - 2014. – 508 p.
8. Васильев А. Н. Mathcad 13 на примерах. – СПб.: БХВ-Петербург, 2006. – 528с.
9. Воскобойников Ю. Е. [и др.] Основы вычислений и программирования в пакете MathCAD : учеб. пособие / под ред. Ю. Е. Воскобойникова; Новосиб. гос. архитектур.-строит. ун-т (Сибстрин). – Новосибирск: НГАСУ (Сибстрин), 2012. – 212 с.

10. Половко А. М., Ганичев И. В. Mathcad для студента. – СПб.: БХВ-Петербург, 2006. – 336с.
11. Очков В. Ф. MathCAD 14 для студентов, инженеров и конструкторов.- СПб.: БХВ-Петербург, 2007. - 368с.
12. Дьяконов В., Круглов В., Математические пакеты расширения MATLAB. Специальный справочник.- СПб.: Питер, 2001. - 480 с.
13. Лазарев Ю. Ф., Початки програмування у середовищі MatLAB: Навч. посібник. - К.: Корнійчук, 1999. - 160 с.
14. Кетков Ю. Л., Кетков А. Ю., Шульц М. М., MATLAB 7: программирование, численные методы. - СПб.: БХВ-Петербург, 2005. - 752с .
15. Амос Гилат, MATLAB. Теория и практика. 5-е изд. / Пер. с англ. Смоленцев Н. К. – М.: ДМК Пресс, 2016. – 416 с.: ил.
16. Лазарев Ю. Ф. Моделирование на EOM. Навчальний посібник. - К.: Політехніка, 2007. - 290 с.

## Додаток А. Основні функції Matlab

### Алгебраїчні оператори та функції

+	– додавання	<code>(plus (a,b)) ;</code>
-	– віднімання	<code>(minus (a,b)) ;</code>
*	- множення	<code>(times (a,b)) ;</code>
/	– ділення ліворуч	<code>(mrdivide (a,b)) ;</code>
\	– ділення праворуч	<code>(mldivide (a,b)) ;</code>
^	– зведення в ступінь	<code>(power (a,b)) .</code>

### Тригонометричні та гіперболічні функції

`sin(Z)`, `cos(Z)`, `tan(Z)`, `cot(Z)` – в тригонометричні;

`sinh(Z)`, `cosh(Z)`, `tanh(Z)`, `coth(Z)` – гіперболічні;

`asin(Z)`, `acos(Z)`, `atan(Z)`, `acot(Z)` – зворотні тригонометричні (в радіанах, в діапазоні від  $-\pi/2$  до  $+\pi/2$ );

`asinh(Z)`, `acosh(Z)`, `atanh(Z)`, `acoth(Z)` – зворотні гіперболічні;

`atan2(X,Y)` – арктангенс (кут в діапазоні від  $-\pi$  до  $+\pi$ );

`sec(Z)`, `csc(Z)` – секанс/косеканс;

`sech(Z)`, `csch(Z)` – гіперболічний секанс/косеканс;

`asec(Z)`, `acsc(Z)` – арксеканс/косеканс;

`asech(Z)`, `acsch(Z)` – зворотний гіперболічний секанс/ косеканс;

### Функції обчислення

`exp(Z)` – експонента;

`log(Z)` – натуральний логарифм;

`log10(Z)` – десятковий логарифм;

`sqrt(Z)` – квадратний корінь;

`abs(Z)` – модуль числа.

`log2(Z)` – логарифм по основі 2;

`pow2(Z)` – 2 в ступені Z;

**sqrt**(Z) – квадратний корінь з Z;  
**nthroot**(Z,n) – корінь n-го ступіню з Z;  
**factorial** (Z) – факторіал Z

### Функції приведення

**fix**(Z) – округлення до найближчого цілого в бік нуля;  
**floor**(Z) – округлення до найближчого цілого в від'ємний бік;  
**ceil**(Z) – округлення до найближчого цілого в додатній бік;  
**round**(Z) – звичайне округлення до найближчого цілого;  
**mod**(X,Y) – цілочислене ділення X на Y;  
**rem**(X,Y) – залишок від ділення X на Y;  
**sign**(Z) – знак числа (0 при Z=0, -1 при Z<0, 1 при Z>0).  
**realmax**('тип') – максимальне число "тип»: 'single', 'double'.  
**realmin**('тип') – число "тип»: 'single', 'double'  
**intmax**('тип') – максимальне число цілого типу. (Див. табл. 1.1)  
**single** (X) – приведення аргументу до типу single.  
**double** (X) – приведення аргументу до типу double.  
**isfloat**(X) – перевірка аргументу. Вірно, якщо аргумент дійсний.  
**isinteger** (X) – перевірка аргументу. Вірно, якщо аргумент цілий.

### Спеціальні функції

**besselj**(n, Z) – функція Бесселя першого роду n порядку;  
**bessely**(n, Z) – функція Бесселя другого роду n порядку;  
**besseli**(n, Z) – модифікована функція Бесселя першого роду n порядку;  
**besselk**(n, Z) – модифікована функція Бесселя другого роду n порядку.  
**beta**(Z,W) – бета-функція;



**betainc**(X,Z,W) – неповна бета-функція;  
**gamma**(Z) – гама-функція;  
**gammainc**(Z,A) – неповна гамма-функція;  
**gamma1n**(Z,A) – логарифм гамма-функція;  
**legendre**(n,X) – узагальнена функція Лежандра;  
**ellipj**(U,M) – еліптична функція Якобі;  
**ellipke**(M) – повний еліптичний інтеграл;  
**erf**(X) – функція похибки (гаусов розподіл);  
**erfc**(X) – додаткова функція похибок;  
**erfcx**(X) – нормована додаткова функція похибок;  
**erfinv**(Y) – зворотна функція похибок.

### Інші функції

**gcd**(A,B) – найбільший загальний дільник;  
**lcm**(A,B) – найменше загальне кратне;  
**rat**(X, tol) – число у вигляді раціонального дробу;  
**tic** – запуск таймера;  
**toc** – зупинка таймера, повернення часу з моменту запуску в секундах;  
**clock** – поточний час. Результат – вектор з 6 чисел: рік, місяць, день, година, хвилина, секунда;  
**etime**(t1, t2) – час між моментами, які завдано векторами t2, t1. Аргументи – вектори формату clock;  
**cputime** – час роботи процесора в мілісекундах зі старту пакета;  
**fzero**(fun, x0, op) – чисельний розв'язок нелінійного рівняння fun. Рівняння записується в нормалізованому вигляді. Способи опису аналогічні функціям оптимізації. Аргументи: **x0** – початкова точка, опціональний параметр **op** – структура умов, аналогічна структурі **OPTIMSET** в функціях

оптимізації. Можливі властивості: **Display**, **TolX**, **FunValCheck**, **OutputFcn**, **PlotFcns**. Виклик з двома результатами повертає окрім кореню значення функції в точці кореня. Виклик з трьома результатами повертає додатково код закінчення розрахунку. Можливі значення: 1 нормальне завершення, -1 завершення по умовам функції **OUTPUT**, -3 отримано значення **NaN** чи **Inf**, -4 в процесі розрахунку отримано комплексні значення, -5 процес не сходиться, -6 функція не може визначити знак рівняння. Виклик з чотирма результатами додатково повертає структуру, яка вміщує поля **OUTPUT.funcCount** – кількість ітерацій, **OUTPUT.algorithm** – кількість ітерацій пошуку інтервалу, **OUTPUT.intervaliterations** – кількість ітерацій, що вміщують нульове значення всередині **OUTPUT.iterations**, **OUTPUT.message**;

**QUAD (FUN, A, B, TOL)** – обчислення інтегралу функції **FUN** від **A** до **B** з точністю **TOL** методом парабол. За замовчання точність  $1.e-6$ ;

**DBLQUAD** – обчислення двократного інтегралу. Дія аналогічна дії функції **QUAD**.

Логічні функції

**==** – перевірка на рівність **(eq(a,b))**,

**~=** – перевірка на нерівність **(ne(a,b))**,

**<** – менше **(lt(a,b))**,

**>** – більше **(gt(a,b))**,

**<=** – менше чи дорівнює **(le(a,b))**,

**>=** – більше чи дорівнює **(ge(a,b))**,

**&** – логічне І **(and(a,b))**,

**|** – логічне ЧИ **(or(a,b))**,

**~** – логічне НІ **(not(a,b))**,

**xor(a,b)** – логічне виключаюче ЧИ,

**any(v)** – вірно, якщо всі елементи вектору дорівнюють нулю,

**all(v)** – вірно, якщо всі елементи вектору НЕ дорівнюють нулю,

**bitand(a,b)** – побітове логічне І,

**bitcmp(a,b)** – побітове логічне ЧИ,

**bitset(a,N<,v>)** – встановлює N-й біт аргументу a в 1(y),

**bitshift(a,N)** – зсуває на N позицій аргумент a. N>0 еквівалентно множенню, N<0 – діленню,

**bitget(a,N)** – зчитує N-й біт аргументу,

**bitxor(a)** – побітове логічне виключаюче ЧИ.

### Матричні функції

**zeros(M,N)** – створення матриці розміром (M\*N) з нульовими елементами,

**ones(M,N)** – створення матриці розміром (M\*N) з одиничними елементами,

**eye(M,N)** – створення одиничної матриці розміром (M\*N) ,

**rand(M,N)** – створення матриці розміром (M\*N) з випадкових чисел, які рівномірно розподілені в діапазоні від 0 до 1,

**randn(M,N)** – створює матрицю розміром (M\*N) з випадкових чисел, які розподілені по гаусову закону с нульовим середнім,

**fliplr(A)** – перестановка стовпців матриці A відносно вертикальної осі,

**flipud(A)** – перестановка рядків матриці A відносно горизонтальної осі,

**rot90(A)** – поворот матриці A на  $90^0$  проти годинникової стрілки,

**tril(A)** – формування нижньої трикутної матриці обнуленням елементів

вище головної діагоналі,

**triu(A)** – формування верхньої трикутної матриці обнуленням елементів

нижче головної діагоналі,

**diag**(x, <n>) – формування діагоналі матриці. Якщо x – вектор, то створюється квадратна матриця з вектором x на діагоналі з вказаним номером (діагоналі рахуються від головної n=1 вгору),

**zeros**(M,N) – формування матриці (**zeros** (1, N) – вектор–рядок, **zeros** (N, 1) – вектор–стовпець) з нульових елементів,

**A'** – транспонування матриці,

**det**(A) – визначник матриці на основі трикутного розкладання методом виключення Гауса,

**trace**(A) – слід матриці (сума діагональних елементів),

**rref**(A) – трикутна матриця на основі методу виключення Гаусса,

**inv**(A) – зворотна матриця,

**cross**(a,b) – векторне множення двох трикомпонентних векторів: вектор,

довжина якого дорівнює площі паралелограма з сторонами a, b, перпендикулярний до площини цих векторів

$$a \times b = \begin{vmatrix} i & j & k \\ a_x & a_y & a_z \\ b_x & b_y & b_z \end{vmatrix} = i(a_y b_z - a_z b_y) + j(a_x b_z - a_z b_x) + k(a_x b_y - a_y b_x). \text{ Результатом є}$$

вектор,

**dot**(a,b) – скалярне множення двох векторів: добуток довжин векторів на косінус кута між ними

$$c = |a||b|\cos(\theta) = a_x b_x + a_y b_y + a_z b_z = a^T \cdot b. \text{ Результатом є скаляр,}$$

**size**(a) – кількість рядків та стовбців матриці. Результат – вектор [n, p],

**max**(a), **min**(a) – значення максимального/мінімального елементу вектору. Визначення результату вектором з двох елементів, додає інформацію про індекс елементу з максимальним/мінімальним значенням.

**mean**(a) – обраховує середнє значення елементів вектору,

**sort(a, mode)** – впорядкування вектору як зростаючого (**mode** – 'ascend') чи зменшуючогося (**mode** – 'descend'),

**sum(a)** – сума елементів вектору,

**prod(a)** – множення елементів вектору,

Функції **size**, **max**, **min**, **sort**, **sum**, **prod** можуть бути застосовані до матриць. В цьому випадку відповідні дії проводяться з кожним із стовпців матриці. Кожен стовпець матриці розглядається як вектор. Функція **sort** сортує елементи кожного стовпця матриці. Результатом є матриця того ж розміру. Функції **sum** та **prod** формують вектор–рядок, кожен елемент якого є сумою чи множенням елементів відповідного стовпця матриці.

**roots(P)** – вектор, елементи якого є коренями поліному,

**poly(r)** – вектор коефіцієнтів поліному по заданому вектору його коренів,

**eig(A)** – повертає вектор особистих значень матриці, тобто коренів характеристичного поліному матриці. Якщо виклик має вид **[R,D]=eig(A)**, то в результатом є діагональна матриця D особистих значень та матриця R правих особистих векторів, які задовольняють умові  $A \cdot R = R \cdot D$ ,

**find(X<,K>)** – повертає вектор з номерами елементів аргументу – вектора (матриці), які не дорівнюють нулю. Для матриці нумерація проводиться по стовпцях згори вниз. Виклик **[I,J,V] = FIND(X,...)** повертає вектори I,J номерів стобців та рядків з номерами елементів, які не дорівнюють нулю. Опціональний вектор V є вектором значень аргументу, які не дорівнюють нулю. В якості аргументу може стояти умова. Наприклад, для матриці A запис **find(A > 5)** поверне вектор з номерами елементів, які більші за 5. Опціональний аргумент "K» обмежує виведення "K» елементів, які задовольняють умові,

**reshape(A,M,N)** – вирізає з матриці A матрицю розміром MxN.

