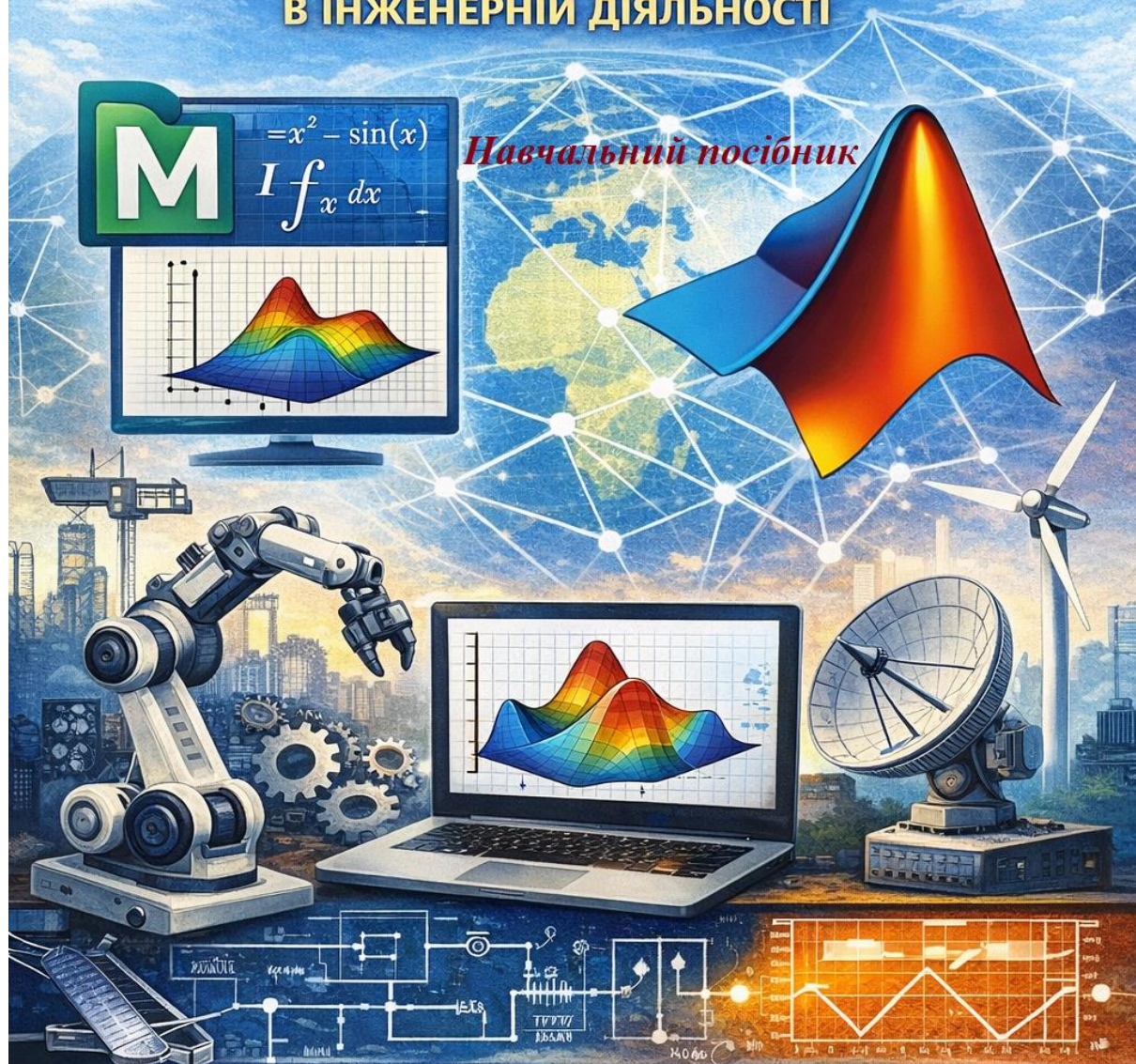


МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
УКРАЇНСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ ЗАЛІЗНИЧНОГО ТРАНСПОРТУ

Д.П. Лабенко, С.О. Змій, А.С. Панченко, О.В. Щебликіна

СУЧАСНІ ПАКЕТИ MATHCAD і MATLAB

ДЛЯ ІМІТАЦІЙНОГО МОДЕЛЮВАННЯ
В ІНЖЕНЕРНІЙ ДІЯЛЬНОСТІ



Харків, 2026



МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
УКРАЇНСЬКИЙ ДЕРЖАВНИЙ УНІВЕРСИТЕТ
ЗАЛІЗНИЧНОГО ТРАНСПОРТУ

СУЧАСНІ ПАКЕТИ MATHCAD І MATLAB
ДЛЯ ІМІТАЦІЙНОГО МОДЕЛЮВАННЯ
В ІНЖЕНЕРНІЙ ДІЯЛЬНОСТІ

Навчальний посібник

Харків 2026

УДК 004.4:519.876 (075.8)

С 91

*Рекомендовано вченою радою Українського державного університету
залізничного транспорту як навчальний посібник
(витяг з протоколу № 8 від 3 липня 2026 р.)*

Рецензенти:

професори О. А. Наконечний (ХУПС),
Є. Зубарєв (НТУ «ХП»)

Авторський колектив:

Д. П. Лабенко, С. О. Змій, А. С. Панченко, О. В. Щєбликіна

С 91 Сучасні пакети MathCAD і MATLAB для імітаційного
моделювання в інженерній діяльності: Навч. посібник /
Д. П. Лабенко, С. О. Змій, А. С. Панченко та ін. – Харків:
УкрДУЗТ, 2026. – 222 с., рис. 153, табл. 75.

ISBN

У навчальному посібнику системно та в повному обсязі подано практичний матеріал для вивчення освітніх компонент «Пакети прикладного програмування» і «Математичні програмні пакети в інженерній діяльності» для здобувачів вищої освіти першого (бакалаврського) рівня спеціальностей G7 (174) «Автоматизація, комп'ютерно-інтегровані технології та робототехніка» і J7 (273) «Залізничний транспорт» усіх форм здобуття освіти.

У посібнику поєднано теоретичні основи математичного та програмного моделювання з прикладною реалізацією задач у середовищах MathCAD і MATLAB.

Крім теоретичних викладок, наведено приклади розв'язання інженерних задач з елементами математичного моделювання, аналізу даних і візуалізації результатів, багато ілюстрацій, продемонстровано інтегративні зв'язки спеціальних дисциплін.

Може бути рекомендований для підготовки здобувачів вищої освіти інших напрямів підготовки, а також для самостійного опанування сучасних програмних засобів інженерних обчислень і моделювання.

УДК 004.4:519.876 (075.8)

ISBN

© Український державний університет
залізничного транспорту, 2026.

ЗМІСТ

ВСТУП	7
Розділ 1. MathCAD – ВІЗУАЛЬНЕ СЕРЕДОВИЩЕ МАТЕМАТИЧНОГО МОДЕЛЮВАННЯ	9
1.1. MathCAD у математичних розрахунках	9
1.1.1. Обчислення значень виразів	11
1.1.2. Визначення і обчислення значень функції в заданій точці	17
1.1.3. Ряди в MathCAD. Сума і добуток ряду	19
1.1.4. Похідна та її обчислення	20
ЗАВДАННЯ ДЛЯ ВИКОНАННЯ	20
1.2. Вхідна мова MathCAD. Управління обчислювальним процесом	24
1.2.1. Вставлення тексту в документ	24
1.2.2. Аналітичне (символьне) і чисельне обчислення виразів	25
1.2.3. Обчислення визначеного інтеграла	33
1.2.4. Обчислення похідної функції $f(x)$	33
ЗАВДАННЯ ДЛЯ ВИКОНАННЯ	35
1.3. Робота з векторами і матрицями в MathCAD	38
1.3.1. Введення матриць і векторів у MathCAD	38
1.3.2. Створення масиву та виведення його значень	39
1.3.3. Векторні і матричні оператори	41
1.3.4. Векторні та матричні функції	44
ЗАВДАННЯ ДЛЯ ВИКОНАННЯ	49
1.4. Робота з масивами, базові операції розв’язання СЛАР	53
1.4.1. Масиви в MathCAD і робота з ними	54
1.4.2. Визначення коренів алгебраїчного рівняння $f(x)=0$	54
1.4.3. Розв’язання систем лінійних рівнянь	60
ЗАВДАННЯ ДЛЯ ВИКОНАННЯ	65
1.5. Побудова і форматування графіків	68

1.5.1. Побудова графіків	68
1.5.2. Основні правила побудови графіків	71
1.5.3. Побудова двовимірних графіків	71
1.5.4. Побудова тривимірних графіків	75
1.5.5. Форматування графіків	79
1.5.6. Крива в просторі	84
1.5.7. Векторні і градієнтні поля	85
ЗАВДАННЯ ДЛЯ ВИКОНАННЯ	86
1.6. Анімація в MathCAD	89
1.6.1. Створення анімаційного кліпу	89
1.6.2. Приклади створення анімаційного кліпу	93
ЗАВДАННЯ ДЛЯ ВИКОНАННЯ	94
1.7. Програмування в системі MathCAD	96
1.7.1. Програмування в пакеті MathCAD	96
1.7.2. Використання програми-функції MathCAD	97
1.7.3. Оператори мови програмування в MathCAD	98
ЗАВДАННЯ ДЛЯ ВИКОНАННЯ	102
Розділ 2. МАТРИЧНА ЛАБОРАТОРІЯ MATLAB	109
2.1. MATLAB у математичних розрахунках. Обчислення математичних виразів	109
2.1.1. Робоче середовище MATLAB	109
2.1.2. Арифметичні операції	111
2.1.3. Змінні та константи.....	112
2.1.4. Алгебраїчні та тригонометричні функції	117
ЗАВДАННЯ ДЛЯ ВИКОНАННЯ	119
2.2. Операції з векторами і матрицями в MATLAB	123
2.2.1. Введення векторів і матриць	123
2.2.2. Створення матриць спеціального виду	125
2.2.3. Функції для розв'язання задач матричної алгебри	129

ЗАВДАННЯ ДЛЯ ВИКОНАННЯ	132
2.3. Використання основних операторів у математичних розрахунках і робота з масивами (матрицями)	135
2.3.1. Блочні матриці	135
2.3.2. Видалення рядків і стовпців	137
2.3.3. Заповнення матриць за допомогою індексації	138
2.3.4. Створення матриць спеціального виду	139
2.3.5. Поелементні операції з матрицями	142
ЗАВДАННЯ ДЛЯ ВИКОНАННЯ	143
2.4. Побудова графіків функцій у MATLAB	147
2.4.1. Побудова графіків	147
2.4.2. Форматування графіків	151
2.4.3. Розбиття графічного вікна на декілька вікон для побудови графіків	155
2.4.4. Побудова анімованих графіків	155
2.4.5. Побудова об'ємних поверхневих діаграм	156
ЗАВДАННЯ ДЛЯ ВИКОНАННЯ	158
2.5. Програмування в середовищі MATLAB	161
2.5.1. Редактор М-файлів	161
2.5.2. Файл-сценарій	163
2.5.3. Файл-функція	165
2.5.4. Оператори мови MATLAB	167
ЗАВДАННЯ ДЛЯ ВИКОНАННЯ	175
2.6. Використання MATLAB для розв'язання задач обчислювальної математики. Розв'язання типових задач алгебри та аналізу	178
2.6.1. Розв'язання систем лінійних рівнянь	178
2.6.2. Знаходження мінімуму заданих функцій	182
2.6.3. Обчислення інтегралів	183
2.6.4. Символьні обчислення і операції Symbolic Math Toolbox ...	184

2.6.5. Спрощення виразів	185
ЗАВДАННЯ ДЛЯ ВИКОНАННЯ	188
2.7. Апроксимація та інтерполяція даних. Методи розв’язання звичайних диференціальних рівнянь	191
2.7.1. Інтерполяція і апроксимація даних	191
2.7.2. Розв’язання звичайних диференціальних рівнянь (ЗДР) ...	196
2.7.3. Розв’язок систем диференціальних рівнянь (СДР)	201
ЗАВДАННЯ ДЛЯ ВИКОНАННЯ	204
2.8. Основні можливості пакета математичного моделювання SIMULINK	207
2.8.1. Початкові відомості про систему Simulink	207
2.8.2. Запуск Simulink	208
2.8.3. Створення (побудова) S-моделі	210
2.8.4. Управління роботою S-моделі	214
2.8.5. Завершення роботи S-моделі	215
2.8.6. Налаштування і виконання створеної моделі	215
2.8.7. Приклади розв’язання найпростіших задач моделювання	216
ЗАВДАННЯ ДЛЯ ВИКОНАННЯ	220
БІБЛІОГРАФІЧНИЙ СПИСОК	222

ВСТУП

Сучасна інженерна діяльність у сфері автоматизації, комп'ютерно-інтегрованих технологій і робототехніки неможлива без використання засобів комп'ютерного моделювання, чисельного аналізу та інтелектуальної обробки даних. Інженер XXI століття повинен не лише володіти фундаментальними математичними знаннями, але і вміти ефективно застосовувати сучасні програмні інструменти для розв'язання прикладних задач аналізу, синтезу та дослідження технічних систем [1-3].

Навчальний посібник «Сучасні пакети MathCAD і MATLAB для імітаційного моделювання в інженерній діяльності» розроблено для здобувачів вищої освіти першого (бакалаврського) рівня спеціальностей G7 (174) «Автоматизація, комп'ютерно-інтегровані технології та робототехніка» і J7 (273) «Залізничний транспорт» усіх форм здобуття освіти в межах освітніх компонент «Пакети прикладного програмування» і «Математичні програмні пакети в інженерній діяльності».

Метою навчального посібника є формування у здобувачів вищої освіти здатності ефективно використовувати сучасні програмні засоби моделювання, аналізу даних і чисельних обчислень у професійній діяльності.

Зміст посібника орієнтований на практичне застосування сучасних математичних програмних пакетів, зокрема MathCAD і MATLAB, для розв'язання типових інженерних задач: виконання чисельних розрахунків, побудови алгоритмів, моделювання динамічних систем, обробки експериментальних даних, візуалізації результатів, аналізу складних технічних процесів.

Методична концепція посібника ґрунтована на діяльнісному підході та передбачає поєднання теоретичних відомостей із практичними прикладами та завданнями для практичного виконання. Особливу увагу

приділено поетапному формуванню навичок: від базових обчислень і візуалізації до створення комплексних моделей технічних систем.

Отже, навчальний посібник спрямований на формування цілісного уявлення про комп'ютерне моделювання як інструмент інженерного мислення та є важливим елементом професійної підготовки майбутніх фахівців у галузі автоматизації, комп'ютерно-інтегрованих технологій і робототехніки.

Розділ 1

МATHCAD – ВІЗУАЛЬНЕ СЕРЕДОВИЩЕ МАТЕМАТИЧНОГО МОДЕЛЮВАННЯ

1.1. MATHCAD у математичних розрахунках

MathCAD – система візуальних математичних розрахунків. Основна ідея MathCAD полягає в тому, що обчислювані вирази записують у візуальній формі, максимально наближеній до математичного запису, звичного для людини. Використовують принцип WYSIWYG (What You See Is What You Get – «що бачите, те й отримуєте») [1-3].

Із запуском системи MathCAD з'являється основне вікно (рис. 1.1), яке зазвичай містить три панелі (стандартну, форматування тексту, математичну, а може бути і більше) і робочу область з автоматично створеним аркушем обчислень.

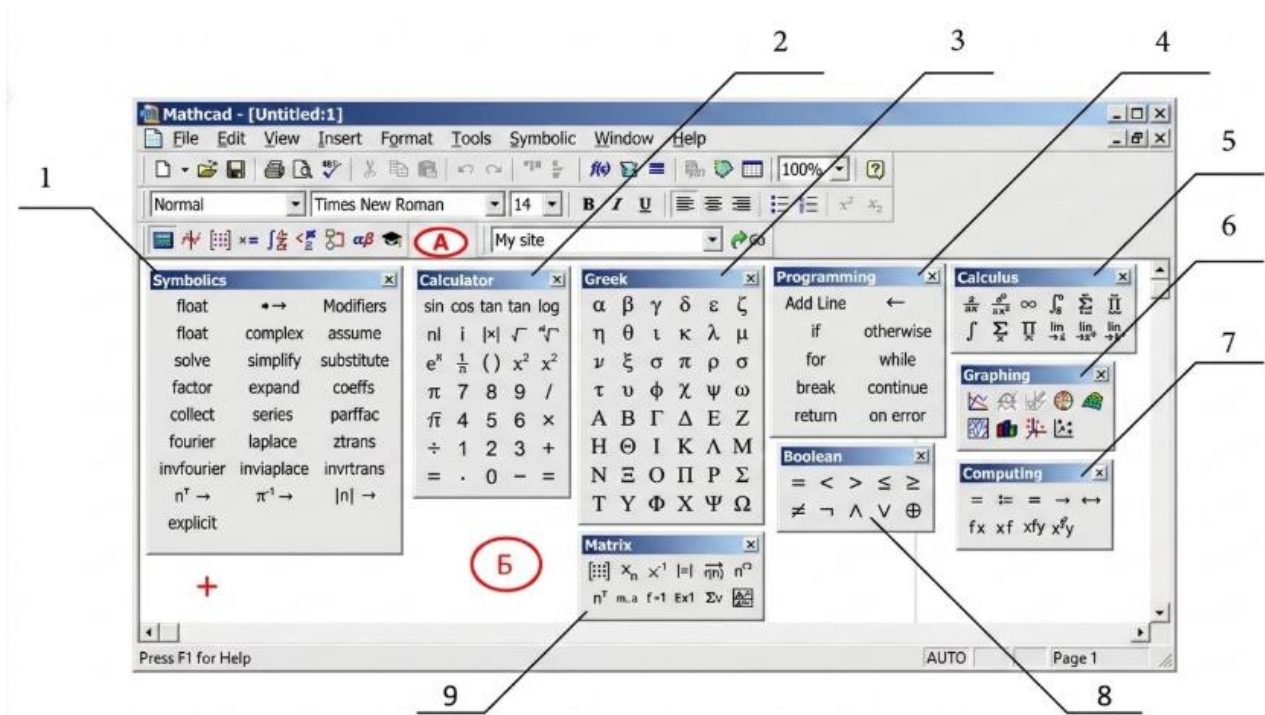


Рис. 1.1. Основне вікно MathCAD

На математичній панелі (А) знаходяться кнопки, із натисканням на які відкриваються додаткові панелі з шаблонами введення різних виразів (табл. 1.1) [1-3].

На аркуш обчислень (Б) записують усі вирази і формули, із якими працюють, також сюди можна вставляти текст і графічні зображення.

Будь-яка формула може бути записана в будь-якому місці аркуша обчислень. Необхідно зафіксувати покажчик миші в передбачуваній точці введення формули, там має з'явитися покажчик введення – хрест червоного кольору (+).

Таблиця 1.1

Вміст математичної панелі

Кнопка	Назва	Призначення
1- 	Символьна (Symbolic)	панель символічних обчислень (різного роду аналітичні перетворення)
2 - 	Калькулятор (Calculator)	панель калькулятора (знаки деяких основних функцій і операцій)
3- 	Грецька (Greek)	панель символів грецького алфавіту (символи грецького алфавіту)
4 - 	Програмування (Programming)	панель програмування (елементи програмування, оператори)
5 - 	Обчислення (Calculus)	панель математичного аналізу (шаблони операцій інтегрування, диференціювання, меж тощо)
6 - 	Графіки (Graph)	панель графіків (графіки функцій, поверхонь, діаграм тощо)
7 - 	Обчислення (Evaluation)	панель обчислень (оператори присвоювання і обчислення)
8 - 	Логічний (Boolean)	панель логічних операцій (логічні операції)
9 - 	Матриця (Matrix)	панель матриць (операції матричного аналізу)

MathCAD інтегрує в собі три редактори: формульний, текстовий і графічний. Для запуску першого достатньо встановити курсор миші в будь-якому вільному місці вікна редагування і натиснути лівою клавішею миші.

1.1.1. Обчислення значень виразів

1.1.1.1. Змінні та константи. Найпростіші обчислення можна виконати, використовуючи знак виведення результатів обчислень «=» (дорівнює). Досить просто ввести з клавіатури необхідний вираз, використовуючи стандартний математичний запис, і натиснути на клавішу «=».

Працюючи з виразами, які необхідно обчислити, можна використати математичні оператори. У табл. 1.2 показані оператори обчислень.

Таблиця 1.2

Оператори обчислень

Оператор	Клавіатура
Додавання	+
Віднімання	-
Множення	*
Ділення	/
Піднесення до ступеня	^ SHIFT +6
Корінь квадратний	\
Факторіал	!
Модуль	SHIFT + \

Наприклад, $15 - \frac{4}{18.9} = 14.788$.

Змінні можна використовувати у виразах MathCAD так само, як і числові константи. Імена змінних можуть включати латинські і грецькі літери, цифри, знаки підкреслення і відсотка, а також допоміжний індекс. Малі та великі літери розрізняються, тому, наприклад, x і X – дві різні змінні.

Щоб *визначити змінну*, достатньо просто присвоїти їй значення. Є суворе правило порядку запису змінних і виразів зі змінними: якщо в деякому виразі використана змінна, то ця змінна має бути визначена на аркуші MathCAD або вище за вираз, або в тому самому рядку, але лівіше (рис. 1.2). Недотримання цього правила – одна з найбільш частих помилок.

$x := 5 \quad Y := 7$ $M := \begin{pmatrix} x & Y \\ Y & x \end{pmatrix}$ $M = \begin{pmatrix} 5 & 7 \\ 7 & 5 \end{pmatrix}$ а	$M := \begin{pmatrix} x & Y \\ Y & x \end{pmatrix}$ $x := 5 \quad Y := 7$ $M = \blacksquare$ б
---	---

Рис. 1.2. Приклади правильної (а) і неправильної (б) послідовностей розташування формул зі змінними

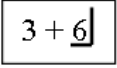
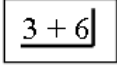
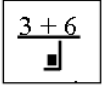
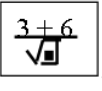
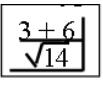
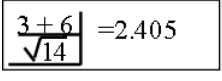
Значення змінних можна змінити в будь-який момент, відредагувавши відповідну формулу. При цьому зазвичай автоматично будуть перераховані всі формули, які безпосередньо або опосередковано залежать від цієї змінної.

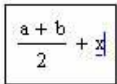
1.1.1.2. Робота з формульними блоками. Робота з формульним блоком – це введення або редагування формул математичних перетворень. Формули вводять безпосереднім набором цифр, імен змінних та операторів. Потрібно враховувати, що в пакеті MathCAD введення математичних операторів має певні особливості:

- MathCAD автоматично вставляє пробіли до та після арифметичних знаків;
- оператор множення вводять як символ зірки « * », а на екрані він має форму крапки «.»;

- оператор ділення вводять як символ похилої лінії «/», а на екрані буде створена двоповерхова конструкція з горизонтальною лінією розподілу;
- оператор піднесення до ступеня вводять символом «^», а на екрані він має вигляд верхнього індексу;
- у звичайному форматі десяткові числа мають не більше трьох цифр дробової частини;
- для розподілу цілої та дробової частин застосовують крапку;
- комбінований оператор присвоювання вводять одним символом «:=».

Приклад. Розрахувати $\frac{(3+6)}{\sqrt{14}}$.

Після введення символів $3 + 6$ на екрані з'явиться: . Курсор введення має вигляд кута, горизонтальна частина якого підкреслює лише цифру 6. Це означає, що наступні оператори можна розташувати тільки праворуч цифри 6 і вони стосуються лише цифри 6. Завдання вимагає ділення суми $3+6$. Для цього горизонтальну частину курсору треба продовжити, натиснувши клавішу «пробіл». На екрані буде . Тепер можна ввести символ ділення «/». На екрані . Символ курсору з'являється у знаменнику і дає змогу ввести шаблон квадратного кореня. Наприклад, введенням символу кореня з палітри **Арифметичні інструменти** (Calculator) або введенням символу «\». На екрані . Після цього вводять цифри 1 і 4 і натискають клавішу «пробіл», поки курсор не підкреслить усю формулу. На екрані . Тепер натискають клавішу «=» і одержують результат . Виведення курсору за межі формульного блока завершує операцію.

Під час введення або редагування формули MathCAD обводить її рамкою введення .

Із набором виразу можна послідовно скасовувати зроблені дії, натискаючи **Ctrl-Z**.

Ще декілька прикладів наведені в табл. 1.3.

Таблиця 1.3

Приклади проведення обчислень

Введення символів	Результат на екрані
1.234*2.345=	$1.234 \cdot 2.345 = 2.894$
1/7 =	$\frac{1}{7} = 0.143$
cos (p (ctrl +G) * 2/3) =	$\cos\left(\pi \cdot \frac{2}{3}\right) = -0.5$
e (shift)+^ 2 =	$e^2 = 7.389$

MathCAD інтегрує в собі три редактори: формульний, текстовий і графічний. Для запуску першого досить встановити курсор миші в будь-якому вільному місці вікна редагування і натиснути лівою клавішею миші. Наприклад, якщо потрібно обчислити добуток членів деякого ряду чисел, то потрібно в панелі **Обчислення** знайти відповідну піктограму і вивести шаблон цієї операції на поле документа .

Після заповнення відповідних позицій шаблону і введення знака рівності = отримаємо результат

$$\prod_{x=1}^{20} \sin(x) \cdot \frac{e}{x} = -2,628 \cdot 10^{-15}$$

У табл. 1.4 наведені операції, які виведені на панель **Обчислення** (*Calculus*), і клавіша, за допомогою якої їх викликають [1-3].

Таблиця 1.4

Операції панелі **Обчислення** (Calculus)

Пікто- грама	Функція	Приклад	Клавіші
	Перша похідна	$\frac{d}{dt} \sin(t) \rightarrow \cos(t)$	Shift+ /
	Похідна n-го порядку	$\frac{d^3}{dt^3} t^5 \rightarrow 60 \cdot t^2$	Ctrl+Shift+ /
	Символ нескінченності	$\int_1^{\infty} x \, dx$	Ctrl+Shift+Z
	Визначений інтеграл	$\int_0^{\pi} \sin(x) \, dx = 2$	&
	Підсумовування	$i := 1 \dots 60 \quad z_i := i^2 \quad \sum_{i=1}^{10} z_i = 385$	Ctrl+Shift+4
	Відшукування добутку	$p := 0.1 \quad \prod_{i=1}^5 (p + 2) = 40.841$	Ctrl+Shift+3
	Невизначений інтеграл	$\int \sin(t) \, dt \rightarrow -\cos(t)$	Ctrl+I
	Сума за дискретним аргументом	$i := 1 \dots 60 \quad z_i := i^2 \quad \sum_i z_i = 7.381 \times 10^4$	\$
	Добуток за дискретним аргументом	$i := 1 \dots 5 \quad p := 0.1 \quad \prod_i (p + 2) = 40.841$	#
	Границя	$\lim_{x \rightarrow 0} \frac{\sin(x)}{x} \rightarrow 1$	Ctrl+L
	Права границя	$\lim_{x \rightarrow \frac{\pi}{2}^+} \tan(x) \rightarrow -\infty$	Ctrl+A
	Ліва границя	$\lim_{x \rightarrow \frac{\pi}{2}^-} \tan(x) \rightarrow \infty$	Ctrl+B

1.1.1.3. Запис виразу. Слід звернути увагу на те, що, використовуючи опцію **Результат** (Number) із меню **Format**, можна управляти розрядністю виведених результатів обчислень. *MatCAD розрізняє малі та великі літери в іменах змінних.*

Для проведення циклічних обчислень із цілочисельною керуючою змінною циклу використовують таку конструкцію:

Ім'я змінної: = Nпоч... Nкін.

Тут знак «...» вводять вибором інструменту  із панелі **Матриця** (рис. 1.3).

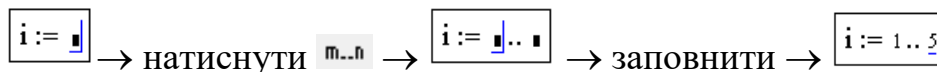


Рис. 1.3. Алгоритм запису ранжованої змінної

$N_{\text{поч}}$ – початкове значення змінної, $N_{\text{кін}}$ – кінцеве значення змінної. Якщо $N_{\text{поч}} < N_{\text{кін}}$, то крок зміни змінної дорівнює **+1**, а якщо $N_{\text{поч}} > N_{\text{кін}}$, то **-1**. Змінні такого типу в системі MathCAD називають змінними із заданими межами вимірювання, або **ранжованими змінними**. Крок зміни можна задати будь-яким, використовуючи іншу конструкцію завдання таких змінних:

Ім'я змінної: = Nпоч, Nнаст... Nкін,

де $N_{\text{наст}}$ – наступне за $N_{\text{поч}}$ значення змінної. Крок у цьому випадку дорівнює $N_{\text{наст}} - N_{\text{поч}}$.

Цикли, реалізовані за допомогою змінних із заданими межами зміни, показані на рис. 1.4.

k := 1..5	i := 5..1	t := -2,-1.5..0	f := 1,0.5..-1																				
k =	i =	t =	f =																				
<table><tr><td>1</td></tr><tr><td>2</td></tr><tr><td>3</td></tr><tr><td>4</td></tr><tr><td>5</td></tr></table>	1	2	3	4	5	<table><tr><td>5</td></tr><tr><td>4</td></tr><tr><td>3</td></tr><tr><td>2</td></tr><tr><td>1</td></tr></table>	5	4	3	2	1	<table><tr><td>-2</td></tr><tr><td>-1.5</td></tr><tr><td>-1</td></tr><tr><td>-0.5</td></tr><tr><td>0</td></tr></table>	-2	-1.5	-1	-0.5	0	<table><tr><td>1</td></tr><tr><td>0.5</td></tr><tr><td>0</td></tr><tr><td>-0.5</td></tr><tr><td>-1</td></tr></table>	1	0.5	0	-0.5	-1
1																							
2																							
3																							
4																							
5																							
5																							
4																							
3																							
2																							
1																							
-2																							
-1.5																							
-1																							
-0.5																							
0																							
1																							
0.5																							
0																							
-0.5																							
-1																							

Рис. 1.4. Приклад ранжованих змінних

Увага! MathCAD читає і виконує введені вирази зліва направо і зверху вниз.

1.1.2. Визначення і обчислення значень функції в заданій точці

Довільні залежності між вхідними і вихідними параметрами задають за допомогою функцій. Функції набувають форми набору параметрів і повертають значення, скалярне чи матричне. У формулах можна використовувати стандартні вбудовані функції, а також функції, визначені користувачем. Щоб використати функцію у виразі, треба вказати вхідні параметри в дужках після імені функції.

1.1.2.1. Вмонтовані функції. У MathCAD великий вибір вмонтованих функцій, які згруповані за категоріями. Імена стандартних математичних функцій ($\ln$, $\log$, e^x , x^2 , x^n , $\sqrt{x}$, $\sqrt[n]{x}$, $n!$, $\sin$, $\cos$, $\tan$) можна ввести з панелі інструментів **Калькулятор**. Інформацію про інші функції можна знайти в довідковій системі. Вставити у вираз стандартну функцію можна за допомогою команди **Вставка → Функція** (або кнопка  на панелі інструментів). У діалоговому вікні **Вставка функції**, що з'явилося, (рис. 1.5) ліворуч вибирають категорію, до якої належить функція, а праворуч – конкретну функцію. У нижній частині вікна буде видана інформація про вибрану функцію. У вікні також є кнопка для одержання довідки про вибрану функцію. Із введенням функції через це діалогове вікно автоматично будуть додані дужки і маркери для заповнення значеннями параметрів [1-3].

1.1.2.2. Функції, задані користувачем. Обчислення функції в точці. Для визначення функції однієї змінної потрібно ввести з клавіатури ім'я функції з аргументом у круглих дужках, знак присвоювання (для введення знака присвоювання потрібно натиснути на клавіатурі комбінацію клавіш **Shift +:** або натиснути по кнопці $:=$ панелі **Обчислення**) і праворуч від нього – вираз для обчислення функції [1-3].

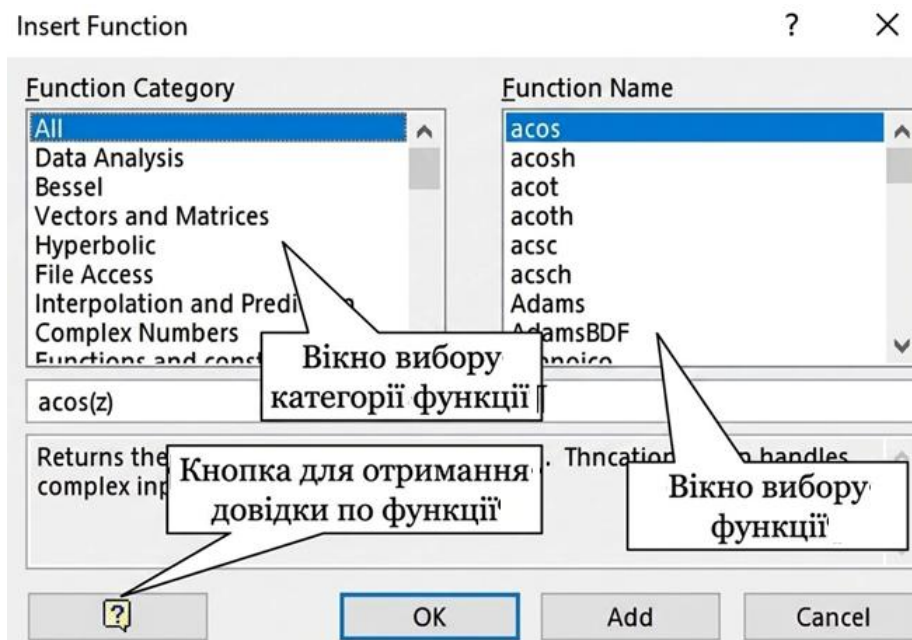


Рис. 1.5. Діалогове вікно *Вставка функції*

Для обчислення значення функції в точці потрібно визначити функцію, а потім ввести з клавіатури ім'я функції, вказати в дужках значення

аргументу, ввести знак дорівнює: $f(x) := \frac{\sin(x)}{x} \quad f(1.5) = 0.665$.

Побудова таблиці значень функції. Для побудови таблиці значень функції необхідно використовувати ранжовані змінні. Для цього визначають дискретні значення аргументу x на відрізку $[a, b]$ із кроком h :

$$x := a, a + h..b.$$

Наприклад, обчислити значення функції $f(x) = x - 5$ на відрізку $[0, 5]$ із кроком 1.

$$x := 0..5$$

$$f(x) := x - 5$$

$f(x) =$

-5
-4
-3
-2
-1
0

MathCAD дає змогу не ставити крок, якщо він дорівнює ± 1 .

1.1.3. Ряди в Mathcad. Сума і добуток ряду

Сума членів ряду позначена в Mathcad так само, як і в класичній математиці, символом $\sum$. Аналогічно добуток членів ряду позначено символом $\prod$. Разом із тим потрібно розрізняти два різновиди обчислення сум і добутоків членів ряду, передбачених у Mathcad. Відповідні шаблони знаходяться на панелі **Обчислення** [1-3].

У першому випадку шаблони містять по чотири маркери, призначення яких відповідають прийнятим у математиці правилам. Приклад запису суми (добутку) членів ряду наведено на лістингу (рис. 1.6).

Обчислення суми ряду

$$\sum_{i=1}^{\infty} \frac{1}{(k+1) \cdot (k+2)} \rightarrow \frac{1}{2} \quad \sum_{k=1}^{\infty} \frac{x^k}{k!} \rightarrow e^x - 1$$

Обчислення суми ряду для ранжированих змінних

$$\sum_{n:=1..5} \frac{n}{10 \cdot n + 1} = 0.479$$

$$\sum_{n:=1..5} \frac{n}{10 \cdot n + 1} \rightarrow \frac{796135}{1663739} = 0.479$$

а

Обчислення добутку ряду

$$\prod_{i=1}^5 \left(\frac{\pi \cdot i}{i+1} \right) \rightarrow \frac{\pi^5}{6} = 51.003$$

Обчислення добутку для ранжированого ряду

$$\prod_{k:=1..5} \frac{1}{k} \rightarrow \frac{1}{120}$$

б

Рис. 1.6. Обчислення суми (а) і добутку ряду (б)

У другому випадку в MathCAD шаблони містять лише два маркери, а сама операція призначена для так званих ранжованих змінних. Ім'я ранжованої змінної (**k**) вказано в нижньому індексі, а її діапазон заданий

додатково. Якщо змінна не визначена, то її вибирають з умови її зміни від **1** до поточного значення змінної **k**.

1.1.4. Похідна та її обчислення

Вираз для похідної функції в MathCAD можна знайти двома способами: за допомогою панелі інструментів і через меню символічних операцій *Символіка-Змінна-Диференціювати*.

Перший спосіб (рис. 1.7, а). На панелі **Обчислення** вибрати інструмент $\frac{d}{dx}$ (*Похідна*). На місце маркера ввести функцію та змінну диференціювання. На панелі **Обчислення** вибрати інструмент $\rightarrow$ (*Обчислити символічно Ctrl+.*). Система виведе значення похідної.

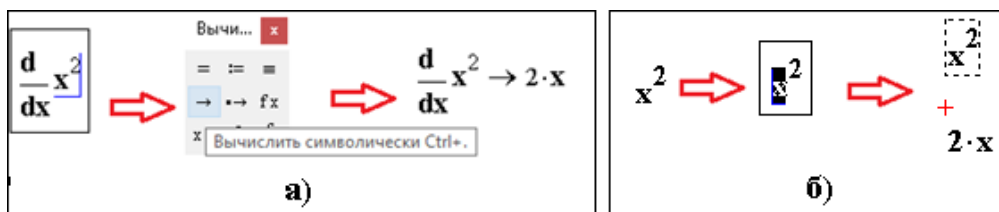


Рис. 1.7. Обчислення похідної функції

Другий спосіб (рис. 1.7, б). Ввести функцію. Виділити змінну на панелі інструментів, виконати команду *Символіка→Змінна→Диференціювати*.

ЗАВДАННЯ ДЛЯ ВИКОНАННЯ

Завдання 1. Обчислити значення числових виразів (табл. 1.5).

Варіанти для завдання 1

Варіант	Вираз
1	$\left(4,5 \cdot 1\frac{2}{3} + 3,75\right) \cdot \frac{7}{135} + \left(\frac{196}{225} - \frac{7,7}{24\frac{3}{4}}\right) + 0,695 : 1,39$
2	$\left\{ \frac{8,8077}{20 - \left[28,2 : 13,333 \cdot 0,3 + 0,0001\right] \cdot 2,004} + 4,9 \right\} \cdot \frac{5}{32}$
3	$\frac{23,1 \cdot \sqrt[3]{3 \cdot \pi - \frac{\ln(54)}{16}}}{5 - \pi^{4!}}$
4	$e^{\sqrt{\frac{22-12,7}{44,8}}} - e^{\pi \left[\left(\frac{1}{3}\right)^6 - 2 \right]}$
5	$\left(\frac{\cos\left(\frac{2 \cdot \pi}{6}\right) \cdot \sin\left(\frac{\pi}{3}\right)}{\tan\left(\frac{2 \cdot \pi}{9}\right) - \log(44)} \right)^3$
6	$\frac{\left(\frac{1}{1!} + \frac{1}{2!} + \frac{1}{3!} + \frac{1}{4!}\right)}{1 - \frac{1}{2} + \frac{1}{3} - \frac{1}{4}}$
7	$e^{\sqrt{\frac{22-12,7}{44,8}}} - \sqrt[3]{3 \cdot \pi - \frac{\ln(54)}{16}}$
8	$\frac{\left(4,5 \cdot 1\frac{2}{3} + 3,75\right) \cdot \frac{7}{135}}{20 - \left[28,2 : 13,333 \cdot 0,3 + 0,0001\right] \cdot 2,004} - \left(0,5 + \frac{1}{3} - \frac{5}{12}\right)$
9	$\left(\frac{0,216}{0,15} + \frac{2}{3} : \frac{4}{15}\right) + \left(4,5 \cdot 1\frac{2}{3} + 3,75\right) \cdot \frac{7}{135}$
10	$\frac{\left[\left(40\frac{7}{30} - 38\frac{5}{12}\right) : 10,9 + \left(\frac{7}{8} - \frac{7}{30}\right) \cdot 1\frac{9}{11}\right] \cdot 4,2}{0,008}$

Завдання 2. Розв'язати квадратне рівняння $ax^2 + bx + c = 0$ для значень a, b, c із табл. 1.6 використовуючи формулу $x_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$.

Таблиця 1.6

Варіанти завдання 2

Варіант	Значення		
	a	b	c
1	2	-8	-3
2	-1,1	-7	1
3	1	5	-1,18
4	-1,3	-6	1,2
5	-7	15	2

Варіант	Значення		
	a	b	c
6	1	-5	23
7	0,5	-2,1	-3
8	4	12	-54
9	18	-1	0,5
10	-9	-6	-2

Завдання 3. Обчислити значення функції в точці та побудувати таблицю значень функції.

1. Дано $x = 1,5$. Обчислити значення функцій $f(x)$ за даними табл. 1.7.
2. Обчислити значення функцій $f(x)$ для 10 значень x , що змінюється в діапазоні від 1 до 3, за даними табл. 1.7.

Таблиця 1.7

Варіанти завдання 3

Варіант	a)	b)	c)
1	2	3	4
1	$f(x) = \frac{2}{3}x^3 - \frac{3}{5x^5} - \frac{7}{\sqrt{x}}$	$f(x) = (\ln x - 5)(\operatorname{tg} x + \sqrt{x})$	$f(x) = \ln(e^x - x^3)$
2	$f(x) = \frac{5}{3}x^3 - \frac{2}{3x^6} + \frac{1}{\sqrt{x}}$	$f(x) = (2 + \sqrt{x})(\operatorname{ctg} x - e^x)$	$f(x) = (3^x - \ln x)^2$
3	$f(x) = 2x^3 - \frac{1}{5x^5} + \frac{4}{\sqrt[4]{x^7}} + 3$	$f(x) = \frac{e^x - \operatorname{tg} x}{\sin x + 2^x}$	$f(x) = \arccos \sqrt{5^x - 1}$
4	$f(x) = \frac{2}{7}x^7 - \frac{1}{4x^4} + \frac{5}{3\sqrt{x}} + 3$	б) $f(x) = \frac{\operatorname{tg} x + \sqrt{x}}{3^x - \sin x}$	$f(x) = \ln(x^3 + e^x - 2)$

1	2	3	4
5	$f(x) = 5x^3 - \frac{3}{4x^4} - 7\sqrt[5]{x^3} - 2,$	$f(x) = \frac{6^x - \cos x}{\operatorname{tg} x + \sqrt{x^3}},$	$f(x) = e^{2x} + 3x \cdot \operatorname{tg} 2x.$
6	$f(x) = \frac{2}{5}x^5 - \frac{3}{4x^4} - 6\sqrt[3]{x} - 4$	$f(x) = \frac{\ln x - e^x}{\operatorname{ctg} x - \cos x}$	$f(x) = \arcsin \sqrt{3^x - \frac{2}{x}}.$
7	$f(x) = 6x^5 - \frac{5}{3x^3} + \frac{6}{\sqrt[4]{x^3}} + 2.$	$f(x) = \frac{3^x + \cos x}{\ln x - \sqrt{x}}$	$f(x) = \operatorname{arctg} \sqrt{x^2 + 1}.$
8	$f(x) = 4x^5 - \frac{7}{4x^4} - \frac{3}{\sqrt[3]{x^2}} + 2$	$f(x) = \frac{\sin x - \cos x}{3^x - \ln x}$	$f(x) = \cos(2x^2 + 3)$
9	$f(x) = \frac{3}{5}x^5 - \frac{1}{2x^4} - \frac{2}{\sqrt[4]{x^3}} + 7$	$f(x) = \frac{5^x - \ln x}{\cos x - 3}$	$f(x) = \sqrt[4]{x^2 + \ln x}$
10	$f(x) = \frac{2}{3}x^3 - \frac{1}{3x^9} + \frac{5}{\sqrt[5]{x^3}} - 6$	$f(x) = (1 - x^2)(\operatorname{tg} x + 3^x).$	$f(x) = \sqrt{x^5 + \sin 5x}.$

Завдання 4. Обчислити суму ряду за даними табл. 1.8 двома способами.

Таблиця 1.8

Варіанти завдання 4

Варіант	Вираз	Варіант	Вираз
1	$\sum_{n=1}^{100} \frac{1}{9n^2 + 3n - 20}$	6	$\sum_{n=1}^{100} \frac{3n - 2}{n(n+1)(n+2)}$
2	$\sum_{n=1}^{100} \left(\frac{n+1}{2n-3} \right)^{n^2}$	7	$\sum_{n=1}^{100} \frac{9}{9n^2 + 21n - 8}$
3	$\sum_{n=1}^{100} \frac{3}{9n^2 - 3n - 2}$	8	$\sum_{n=1}^{100} \frac{n(n^2 - 1)}{n!}$
4	$\sum_{n=3}^{100} \frac{(n-1)(n+1)(n-2)}{8n - 10}$	9	$\sum_{n=2}^{100} \frac{1}{n(n^2 - 1)}$
5	$\sum_{n=1}^{100} \frac{n!}{n^{n-1}}$	10	$\sum_{n=1}^{100} \left(\frac{2n+1}{3n-2} \right)^{n^2}$

Завдання 5. Похідна та її обчислення. Визначити похідну функції за даними табл. 1.7 двома способами.

Завдання 6. Обчислити самостійно значення визначеного інтеграла за даними табл. 1.9.

Таблиця 1.9

Варіанти завдання 6

Варіант	Вираз	Варіант	Вираз
1	$\int_0^{\pi} x^3 \sin x dx$	6	$\int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} \frac{dx}{1 + \cos x}$
2	$\int_1^e \ln^2 x dx$;	7	$\int_0^1 x e^{-x} dx$
3	$\int_0^{\pi/2} \frac{\sin(x) dx}{5 + 3 \sin(x)}$	8	$\int_{-\frac{\pi}{2}}^{\frac{\pi}{4}} \frac{\cos^2 x dx}{\sqrt[3]{\sin x}}$
4	$\int_{-2}^0 (x^2 + 2) e^{x/2} dx$	9	$\int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} \cos t \sin \left(2t - \frac{\pi}{4} \right) dt$
5	$\int_0^{e-1} \ln x + 1 dx$	10	$\int_2^9 \frac{x dx}{\sqrt[3]{x-1}}$

1.2. Вхідна мова MATHCAD. Управління обчислювальним процесом

1.2.1. Вставлення тексту в документ

Текстові блоки дають змогу створювати в документі пояснення, тобто робити документ MathCAD більш зрозумілим для читання. Для введення тексту в місці розташування курсору створюють текстовий блок. Тут можливі варіанти:

- текстовий блок вводять командою **Вставка → Текстова область** (Insert→Text Region);

- для введення текстового блока досить ввести символ " (лапки). У прямокутнику, який з'явився, можна вводити текст і редагувати його;

- третій варіант передбачає безпосереднє введення послідовності текстових символів, яка перетворюється в текст тільки після натискання клавіші «Пробіл» (Backspace) [1-3].

Активний текстовий регіон має вигляд прямокутної рамки з трьома чорними маркерами. Перетягування маркера дає змогу змінювати геометричні розміри регіону, не змінюючи розмірів тексту. Текстовий курсор (вертикальна червона риска) визначає місце введення наступного символу. Рамка та маркери зникають із робочого поля після виведення текстового курсору за межі регіону.

1.2.2. Аналітичне (символьне) і чисельне обчислення виразів

Кожен оператор у **MathCAD** означає деяку математичну дію у вигляді символу. Символ будь-якого оператора вводять у потрібне місце документа одним із двох способів:

- натисканням відповідної клавіші на клавіатурі;
- натисканням лівою кнопкою миші на відповідну кнопку на одній із математичних панелей.

Обчислювальні оператори згруповані на панелі **Обчислення (Evaluation)**. Це:

- чисельне виведення (=);
- символьне виведення (→);
- присвоєння (:=);
- глобальне присвоєння (≡) [1-3].

У системі **MathCAD** алгебраїчні обчислення виконують переважно аналітично. Це допомагає суттєво зекономити свій час і сили, затрачені на виконання всіляких перетворень математичних виразів.

1.2.2.1. Символьне обчислення. Символьні обчислення в **MathCAD** можна виконувати:

- за допомогою команд меню;
- допомогою оператора символьного виведення $\rightarrow$, ключових слів символьного процесора та звичайних формул.

Перший спосіб більш зручний тоді, коли потрібно швидко отримати аналітичний результат для одноразового використання, не зберігаючи хід обчислень.

Другий спосіб дає змогу записувати вирази в традиційній математичній формі і зберігати символьні обчислення в документах **Mathcad**. Символьний процесор **Mathcad** може виконувати основні алгебричні перетворення, такі як спрощення виразів, розкладання їх на множники, символьне обчислення суми та добутку [1-3].

Для символьних обчислень за допомогою команд призначено головне меню **Символіка** (Symbolics), яке об'єднує математичні операції, які **Mathcad** вміє виконувати аналітично.

Для реалізації другого способу використовують засоби **Mathcad**, наприклад панелі **Калькулятор**, **Обчислення** тощо, і спеціальну математичну панель (рис. 1.10) **Символіка** (Symbolics). На цій панелі знаходяться кнопки, які відповідають специфічним командам символьних перетворень [1-3].

Розглянемо обидва способи символьних обчислень на простому прикладі розкладання на множники виразу **cos(4x)**.

Перший спосіб.

1. Ввести вираз **cos(4x)**.
2. Виділити його повністю.

3. Вибрати в головному меню пункти **Символіка/Розгорнути** (Symbolics/Expand).

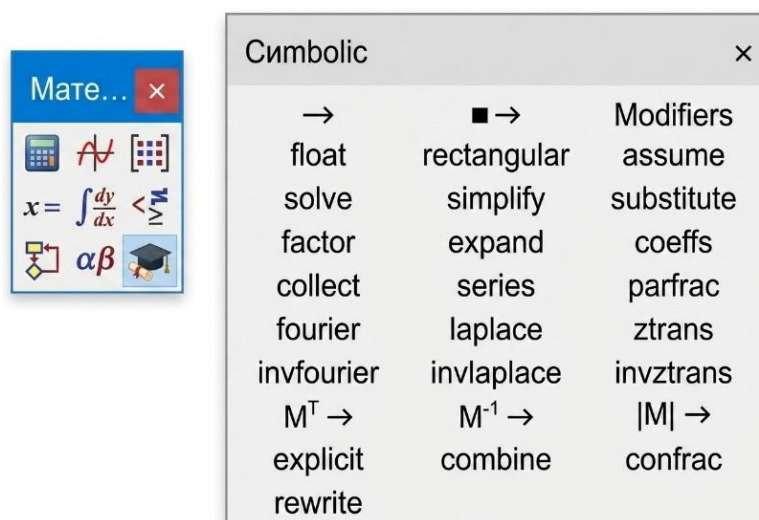


Рис. 1.10. Кнопки панелі **Символи**

Після цього результат розкладання виразу появиться нижче у вигляді ще одного рядка (рис. 1.11).

$$\cos(4 \cdot x) \Rightarrow \boxed{\cos(4 \cdot x)} \Rightarrow \begin{matrix} \cos(4 \cdot x) \\ \cos(x)^4 - 6 \cdot \cos(x)^2 \cdot \sin(x)^2 + \sin(x)^4 \end{matrix}$$

Рис. 1.11. Проведення символічних обчислень через головне меню

Другий спосіб.

1. Ввести вираз **$\cos(4x)$** .
2. Натиснути кнопку **Розгорнути вираз** (Expand) на панелі **Символіка** (Symbolics).
3. Натиснути мишею поза рамкою, і на екрані появиться результат (рис. 1.12).

$$\cos(4 \cdot x) \Rightarrow \boxed{\cos(4 \cdot x) \text{ expand}} \rightarrow \Rightarrow \cos(4 \cdot x) \text{ expand} \rightarrow \cos(x)^4 - 6 \cdot \cos(x)^2 \cdot \sin(x)^2 + \sin(x)^4$$

Рис. 1.12. Проведення символьних обчислень через панель **Символіка**

Не кожен вираз піддається аналітичним перетворенням. Якщо так, то або задача зовсім не має аналітичного розв'язку, або вона є надто складною для символьного процесора **MathCAD**, і як результат буде виведений початковий вираз.

1.2.2.2. Спрощення виразів. Символьний процесор **MathCAD** перетворює вирази так, що вони набувають більш простішого вигляду. При цьому використовують різні арифметичні формули, зведення подібних доданків, тригонометричні тотожності, перерахунок обернених функцій та ін. Щоб спростити вираз за допомогою меню, необхідно:

1. Ввести вираз.
2. Виділити вираз повністю або його частину, яку необхідно спростити.
3. Вибрати команду **Символіка/Спростити** (Symbolics/Simplify).

Для спрощення виразів за допомогою оператора символьного виведення використовують ключове слово *simplify*. Якщо деяким змінним, які входять до виразу, раніше були присвоєні деякі значення, то вони будуть підставлені у вираз із символьного виведенням.

Приклад. Спрощення виразу (рис. 1.13).

$$\frac{a + b - a}{2 \cdot a} \text{ simplify} \rightarrow \frac{b}{2 \cdot a} \quad . \quad \frac{a + b - a}{2 \cdot a} \rightarrow \frac{b}{2 \cdot a}$$

Рис. 1.13. Спрощення виразу різними способами

Найбільш яскравим прикладом можливостей символьного процесора **MathCAD** є аналітичне обчислення границь, похідних, інтегралів, розкладання в ряди, обчислення сум і добутків, а також розв'язання алгебричних рівнянь. Усі ці операції, із виконанням їх засобами меню **Символіка** (Symbolics), знаходяться в його підменю **Змінна** (Variable). Відповідно, потрібне попереднє виділення у виразі змінної, відносно якої буде виконуватись операція. Для цього краще виділити її чорним кольором протягуванням курсору миші через потрібну частину виразу [1-3].

Приклад. Приклади символьного та чисельного обчислення (рис. 1.14).

$$\sum_{n=0}^{100} \frac{1^n}{2^n \cdot n!} = 1.649 \cdot \prod_{n=1}^{\infty} \frac{1}{n+1} \rightarrow 0 \cdot \prod_{n=1}^{\infty} \sqrt{n} \rightarrow \infty$$

Рис. 1.14. Приклади символьного і чисельного обчислення

1.2.2.3. Обчислення виразів із рухомою крапкою. Для організації обчислень із рухомою крапкою передбачене спеціальне ключове слово **float**, яке вводять із панелі **Символьна** (Symbolic) разом із значенням точності виведення результату з рухомою крапкою (рис. 1.15).

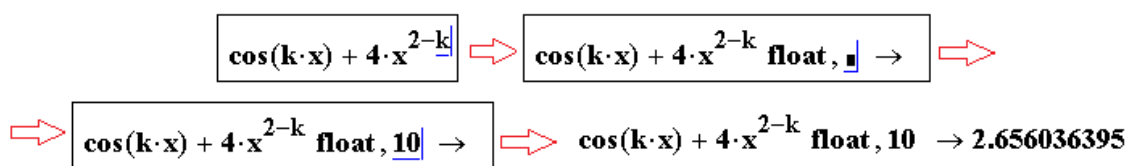


Рис. 1.15. Порядок обчислення виразів із рухомою крапкою

1.2.2.4. Робота з операторами поширених можливостей і вбудованими функціями. У палітрі **Оператори математичного аналізу** (Calculus) зібрані шаблони складних функцій (приклади показано на рис. 1.16).

$$\frac{d}{dx} \quad \int_1^2 x dx$$

Рис. 1.16. Приклади шаблонів диференціювання та інтегрування

Чорні прямокутники в шаблонах необхідно заповнити належною інформацією, виділити формульним курсором усю формулу і введенням символу “=” одержати результат (рис. 1.17).

$$x := 0.1 \quad \frac{d}{dx} x^2 = 0.2 \quad \int_0^1 x^2 dx = 0.333 \quad i := 1..8 \quad \sum_i i = 36 \quad \prod_i i = 4.032 \times 10^4$$

Рис. 1.17. Приклади розрахунків із використанням шаблонів

Пакет MathCAD має чималий перелік вбудованих функцій, список яких викликають клавішею ***f(x)*** – **Вставити Функцію** (Insert Function) стандартної панелі редагування. Функції також можна вводити набором відповідних символів із клавіатури. Приклади наведені на рис. 1.18.

$$x := 1.34 \quad e^x = 3.819 \quad \tan(x) = 4.256 \quad \ln(x) = 0.293$$

Рис. 1.18. Приклади розрахунків за функціями

У табл. 1.8 наведені основні типові вирази та шаблони, які використовують для введення формул із клавіатури.

Введення типових виразів і шаблонів MathCAD із клавіатури

Потрібна дія	Вигляд виразу	Послідовність натискання клавіш
1	2	3
Множення F на X	$F \cdot X$	F*X
Ділення F на X	$\frac{F}{X}$	F/X
Піднесення F до ступеня A	F^A	F^A
Квадратний корінь з y	$\sqrt{y}$	\y
Модуль z. Тут і далі у фігурних дужках будуть перераховані клавіші, які необхідно натискати одночасно. У цьому випадку після натискання клавіші з літерою «z» необхідно одночасно натиснути клавіші «Shift» і «\»	$ z $	z {Shift \}
Поєднане (пов'язане) до z комплексне число	$\bar{z}$	z {Shift "}
Числове обчислення виразу 1 + 2	1+2=	1+2=
Аналітичне обчислення виразу A. Результатом такого обчислення є не число, а формула (наприклад результат обчислення інтеграла)	$A \rightarrow$	A {Ctrl.}
Присвоєння змінній A значення 5	A:=5	A:5
Визначення послідовності F у діапазоні від 10 до 15 із кроком 1. F = 10,11,12,13,14,15	F:=10..15	F: 1; 10
Визначення послідовності F у діапазоні від 2 до 10 із кроком 2. F = 2,4,6,8,10	F:=2,4..10	F: 2 , 4; 10
i-й елемент вектора F	F_i	F [i
Елемент матриці F з індексами (i, j)	$F_{i,j}$	F [i , j
Введення змінної з допоміжним індексом. Тут індекс – тільки частина імені	$F_{\max}$	F.max
Транспонування матриці. Не слід плутати його з піднесенням до степеня	A^T	A {Ctrl 1}
Поелементне множення матриць. Тут клавішу «пробіл» використовують для зміни рівня введення (буде роз'яснено далі)	$\overline{A \cdot B}$	A*B пробел {Ctrl -}
Введення матриці або вектора. Після натискання Ctrl-M відкриється вікно, у якому необхідно вказати кількість рядків (у пункті Rows) і стовпців (у пункті Columns)	$\begin{bmatrix} \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \\ \cdot & \cdot & \cdot \end{bmatrix}$	{Ctrl M}
Корінь n-го степеня з y	$\sqrt[n]{y}$	{Ctrl \} y
Перша похідна	$\frac{d}{dx}$	{Shift /}

1	2	3
Двостороння межа	$\lim_{x \rightarrow a} f(x)$	{Ctrl L}
Невизначений інтеграл	$\int f(x) dx$	{Ctrl I}
Грецькі символи. Вводять літеру латинського алфавіту (зазвичай вимовляють так само, як перша літера назви грецького символу) і натискають Ctrl-G	$\alpha \beta \delta \epsilon \phi \gamma \varphi \lambda$ $\mu \pi \rho \tau \omega \psi$	a b d e f g j l m p r t w y {Ctrl G}

1.2.2.5. Функції користувача. Функції користувача дають змогу спростити формули, зробити їх компактними та зрозумілими для аналізу. У загальному випадку функції мають таку конструкцію:

Ім'я (Список аргументів):= Вираз.

Приклад. $f(x) := x e^x$.

Функції можуть бути без аргументів: $F := \frac{1+2}{3+4} + 5^{0.6}$ F=3.055.

Фактичні аргументи використовують замість формальних:

$f(x) := x \cdot \sin(x)$
 $z := 0.8 \quad f(z) = 0.574$.

Функція сама може бути оператором або аргументом іншої функції. Крім того, змінні у списку формальних аргументів функції є локальними, застосовуваними тільки в цій функції. Можна користуватися змінними, визначеними за межами функції:

$A := 2 \quad f(x) := A \cdot x \cdot e^x \quad \int_0^5 f(x) dx = 1.189 \times 10^3$.

Важливе значення має спосіб визначення глобальної змінної. У пакеті існує оператор глобального присвоювання (три риси) із палітри **Інструменти деяких знаків** (Evaluation). Присвоєння таким оператором дійсне в межах усього документа від початку до кінця.

$$\text{Приклад.} \quad f(x) := A \cdot x \cdot e^x$$

$$A \equiv 10$$

1.2.3. Обчислення визначеного інтеграла

Інтегрування, як і диференціювання, і як більшість інших математичних дій, реалізовано в **Mathcad** за принципом «як пишуть, так і вводять». Щоб обчислити визначений інтеграл, необхідно ввести звичну математичну форму в документ за допомогою панелі **Обчислення** (Calculus), на якій натиснути кнопку із значком інтеграла. У результаті з'явиться шаблон інтеграла з декількома місцезаповнювачами, у які необхідно ввести нижню та верхню межі інтегрування, підінтегральну функцію та змінну інтегрування. Щоб отримати результат інтегрування, необхідно ввести знак рівності або символьної рівності. У першому випадку інтегрування буде виконано чисельним методом, а в другому в разі успішного результату буде знайдено точне значення інтеграла за допомогою символьного процесора **MathCAD** [1-3].

Приклад. Обчислення чисельного та символьного обчислення визначеного інтеграла (рис. 1.19).

$$\int_0^{\pi} e^{-x^2} dx = 0.886 \qquad \int_0^{\pi} e^{-x^2} dx \rightarrow \frac{\sqrt{\pi} \cdot \text{erf}(\pi)}{2} = 0.886$$

Рис. 1.19. Обчислення виразів визначеного інтеграла

1.2.4. Обчислення похідної функції f(x)

Для того щоб аналітично знайти похідну функції **f(x)** в **MathCAD**, необхідно:

1. Задати функцію $f(x)$.
2. Ввести шаблон $\frac{d}{dx}$.
3. У місце заповнювачів оператора диференціювання ввести функцію $f(x)$ та ім'я аргументу x .
4. Ввести оператор символьного обчислення $\rightarrow$ або виконати операцію *simplify*, вибравши її на панелі **Символьна**, і натиснути клавішу Enter (рис. 1.20).

$$\begin{array}{ccc}
 f(x) := \sin(x) \cdot \ln(x) & & \\
 \frac{d}{dx} f(x) \rightarrow \frac{\sin(x)}{x} + \cos(x) \cdot \ln(x) & \quad \frac{d}{dx} f(x) \text{ simplify } \rightarrow \frac{\sin(x)}{x} + \cos(x) \cdot \ln(x) & \\
 \text{a)} & & \text{б)}
 \end{array}$$

Рис. 1.20. Обчислення похідної

Для того щоб обчислити похідну в точці, потрібно попередньо задати значення аргументу. Результатом диференціювання в цьому випадку буде число – значення похідної в цій точці. Якщо результат вдалося відшукати аналітично, то його наводять як числовий вираз, а щоб отримати його у формі числа, достатньо ввести після отриманого виразу символ чисельної рівності «=» (рис. 1.21).

$$\begin{array}{ccc}
 x := 2 & f(x) := \sin(x) \cdot \ln(x) & \\
 \frac{d}{dx} f(x) = \mathbf{0.166} & \quad \frac{d}{dx} f(x) \rightarrow \frac{\sin(2)}{2} + \cos(2) \cdot \ln(2) = \mathbf{0.166} &
 \end{array}$$

Рис. 1.21. Обчислення похідної в точці

Інші приклади проведення операцій наведені на рис. 1.22.

$$\frac{d}{dz}(z \cdot \sin(2 \cdot z)) \rightarrow \sin(2 \cdot z) + 2 \cdot z \cdot \cos(2 \cdot z)$$

$$\int z \cdot \sin(2 \cdot z) dz \rightarrow \frac{\sin(2 \cdot z)}{4} + \frac{z \cdot (2 \cdot \sin(z)^2 - 1)}{2}$$

$$\int_0^{\pi} z \cdot \sin(2 \cdot z) dz \rightarrow -\frac{\pi}{2} \quad \text{порівняйте} \quad \int_0^{\pi} z \cdot \sin(2 \cdot z) dz = -1.571$$

$$\lim_{y \rightarrow 0} \frac{\sin(2 \cdot y)}{y} \rightarrow 2$$

Рис. 1.22. Приклади проведення аналітичного обчислення

ЗАВДАННЯ ДЛЯ ВИКОНАННЯ

Завдання 1. Вставити в документ текстову область (п. 1.2.1).

Завдання 2. Розрахувати аналітично і чисельно для значення $x = N$ (де N – номер за списком у журналі) такі функції:

$$y = \frac{1}{\sqrt[3]{x^2} + \sqrt[6]{x^5}}; \quad y = \frac{1}{\sqrt{2 \cdot \pi}} e^{\frac{-x^2}{2}}; \quad y = \frac{\arctg(x)^2}{2}.$$

Завдання 3. Скопіювати першу із формул завдання 2 і замінити в ній перший підкореневий вираз у знаменнику на ***sin(x)***, а другий операнд у знаменнику на ***ctg(x/2)*** для парних варіантів і перший підкореневий вираз у знаменнику на ***ctg(x/2)***, а другий операнд у знаменнику на ***sin(x)*** для непарних варіантів. Присвоїти отриманий вираз новій змінній. Обчислити для заданого $x = N + 1$ отриманий вираз.

Завдання 4. Обчислити значення виразу з табл. 1.9 для свого варіанта, самостійно вибравши значення змінної x .

Завдання 5. Задати і вивести на екран значення ранжованої змінної x , яка змінюється:

- від 0 до 2π із кроком 1 для парних номерів за списком у журналі;
- від 0 до -2π із кроком -1 для непарних номерів за списком у журналі.

Завдання 6. Визначити функцію $f(x)$ із табл. 1.9 для свого варіанта, обчислити її значення. Значення змінної задати самостійно як ранжовану.

Завдання 7. Вставити в документ текстову область «Робота з математичними формулами».

Завдання 8. Побудувати таблицю значень функції $f(x)$ для свого варіанта з таблиці завдань на відрізку $[1,20]$ із кроком $(1/N)$, вивести значення x і результат на цьому інтервалі.

Завдання 9. Вставити в документ текстову область «Обчислення суми ряду».

Завдання 10. Обчислити аналітично і чисельно суму для заданих рядів із табл. 1.10 (по три підряд на ваш вибір), самостійно вибравши кількість елементів ряду (інтервал зміни значення n).

Завдання 11. Вставити в документ текстову область «Похідна та її обчислення».

Завдання 12. Обчислити аналітично і чисельно похідну функцій $f(x)$ із таблиці завдань для п. 6 (варіант + ще два наступні), самостійно вибравши значення змінної.

Таблиця 1.9

Варіанти завдань

Варі- ант	Для завдання 4. Обчислити значення виразів	Для завдання 6. Визначити функцію $f(x)$ і обчислити її значення
1	2	3
1	$2x^3 - \frac{1}{5x^5} + \frac{4}{\sqrt[4]{x^7}}$	$f(x) = \frac{e^x - \operatorname{tg} x}{\sin + 2^x}$
2	$\frac{2}{5}x^5 - \frac{3}{4x^8} - 7\sqrt[3]{x^{23}} - 1$	$f(x) = \frac{x^5 - e^x}{\operatorname{arctg} x + \sqrt{x}}$
3	$\frac{7}{9}x^9 - \frac{4}{3x^3} + 5\sqrt[5]{x^3} - 1$	$f(x) = \frac{\operatorname{ctg} x - 3}{5^x + \sqrt{x}}$

Продовження табл. 1.9

1	2	3
4	$\frac{2}{5}x^5 - \frac{3}{4x^4} - 6\sqrt[3]{x} - 4$	$f(x) = \frac{\ln x - e^x}{\operatorname{ctg} x - \cos x}$
5	$\frac{2}{7}x^7 - \frac{1}{4x^4} + \frac{5}{3\sqrt{x}} + 3$	$f(x) = \frac{\operatorname{tg} x + \sqrt{x}}{3^x - \sin x}$
6	$4x^5 - \frac{3}{x^3} - \frac{2}{\sqrt[5]{x^3}}$	$f(x) = (\ln x + \operatorname{tg} x)(\sqrt{x} - e^x)$
7	$5x^3 - \frac{3}{4x^4} - 7\sqrt[5]{x^3}$	$f(x) = \frac{6^x - \cos x}{\operatorname{tg} x + \sqrt{x^3}}$
8	$6x^5 - \frac{5}{3x^3} + \frac{6}{\sqrt[4]{x^3}} + 2$	$f(x) = \frac{3^x + \cos x}{\ln x - \sqrt{x}}$
9	$3x^2 - \frac{1}{7x^7} + \frac{3}{\sqrt[3]{x^2}} + 1$	$f(x) = \frac{5^x - \ln x}{\cos x - 3}$
10	$\frac{2}{3}x^3 - \frac{1}{3x^9} + \frac{5}{\sqrt[5]{x^3}} - 6$	$f(x) = (1 - x^2)(\operatorname{tg} x + 3^x)$

Таблиця 1.10

Варіанти завдань

	Для завдання 9. Побудувати таблицю значень на відрізку [1,20] із кроком 1/N (де N – номер ПК)	Для завдання 11. Обчислити суму заданого ряду
1	$f(x) = (\ln x + \sin 3x)^5$	$\frac{\sqrt{n+2} - \sqrt[3]{n}}{\sqrt[3]{n^2+2n}}$
2	$f(x) = \arccos \sqrt{5^x - 1}$	$1 - \frac{1}{(n+1)^{n+1}}$
3	$f(x) = (3^x - \ln x)^2$	$\frac{1}{4^n} + (-1)^n \frac{2^n}{3^{n+1}}$
4	$f(x) = \ln(e^x - x^3)$	$\frac{(-3)^{n+1} + 5n^{n-1}}{5^{n+2}}$
5	$f(x) = (\operatorname{tg} 3x)^5$	$\frac{3n+4}{n(n+1)(n+4)}$
6	$f(x) = \operatorname{arctg} \sqrt{x^2 + 1}$	$\frac{9}{9n^2 + 3n - 20}$
7	$f(x) = \cos(2x^2 + 3)$	$\frac{(-1)^n + 2^n}{4^n}$
8	$f(x) = \sqrt{x^5 + \sin 5x}$	$\frac{3^n + (-2)^n}{6^n}$
9	$f(x) = \sqrt[4]{x^2 + \ln x}$	$\sqrt{n+2} - 2\sqrt{n+1} + \sqrt[5]{n}$
10	$f(x) = \arcsin \sqrt{3^x - \frac{2}{x}}$	$\frac{n^2 + 3n + 2}{n^2 + 3n}$

1.3. Робота з векторами і матрицями в MathCAD

Ознайомлення з особливостями роботи з векторами і матрицями в середовищі MathCAD; порядок проведення простих математичних розрахунків за допомогою операторів MathCAD; ознайомлення із вбудованими функціями MathCAD, які обробляють масиви; набуття практичних навичок роботи в середовищі з векторами і матрицями.

1.3.1. Введення матриць і векторів у MathCAD

1.3.1.1. Масиви в MathCAD. *Масив* (Array) – іменований упорядкований набір даних, кожний із яких має свій порядковий номер – нижній індекс. Індекс указує на елемент масиву, його вводять після імені масиву кнопкою **Нижній індекс** (Subscript)  на панелі інструментів **Матриці**, або гарячою клавішею [(Ліва квадратна дужка), або кнопкою **Нижній індекс** (Subscript)  на панелі інструментів **Форматування** [1-3].

Нумерувати елементи масиву в MathCAD за замовчуванням починають із нуля, тому що саме це значення має вбудована змінна **ORIGIN = 0** (ПОЧАТОК).

Ім'я змінної завжди набирають заголовними літерами. У математиці зазвичай елементи масивів нумерують з одиниці. Це можна зробити і в MathCAD оператором **ORIGIN:= 1** на початку робочого аркуша.

Масив має дві характеристики:

- 1) **розмірність масиву** – кількість нижніх індексів елемента масиву;
- 2) **розмір масиву** – сумарна кількість елементів.

За розмірністю масиви поділяють так:

1) **вектор** – одновимірний масив (розмірність дорівнює одиниці, тобто елементи вектора мають один нижній індекс);

2) **матриця** – двовимірний масив (елементи матриці мають два нижні індекси: перший – номер рядка, другий (через кому) – номер стовпця);

3) **тензор** – багатовимірний масив, або в MathCAD – гніздовий масив.

Елементом масиву може бути число, константа, проста змінна (у тому числі дискретна), масив, функція, вираз, формула.

Масив, який містить як елементи інші масиви, так звані вкладені масиви, називають гніздовим масивом (Nested Array) [1-3].

1.3.2. Створення масиву та виведення його значень

У MathCAD можна створити масив декількома способами.

1.3.2.1. Прямий спосіб введення елементів масиву в шаблон матриці:

➔ на панелі інструментів **Матриця** натиснути кнопку **Вектор або Матриця** (Vector or Matrix) .

➔ у діалоговому вікні **Вставити матрицю** (Insert Matrix) указати кількість **рядків** (Rows) і **стовпчиків** (Columns);

➔ у шаблон матриці, що з'явиться, ввести значення елементів, натискаючи клавішу → **Праворуч** (Right) для переходу в наступне поле введення.

Матриця розміром $n \times 1$, тобто матриця-стовпець із n рядків, є вектором. Дійсно, оскільки другий індекс завжди дорівнює одиниці, то його можна опустити, і, отже, елемент такої матриці завжди буде мати тільки один нижній індекс. Саме тому матриця-стовпець у MathCAD є вектором.

Матриця розміром $1 \times n$, тобто матриця-рядок із n стовпців, тільки виглядає як вектор-рядок, оскільки її елемент обов'язково має два індекси.

1.3.2.2. Застосування дискретної змінної для введення-виведення масиву. Розглянемо це питання на прикладі вектора $v=\{3, 5, 7\}$ (рис. 1.22):

- ➔ (1) - почати індексацію елементів масивів з одиниці;
- ➔ (2) - створити цілочислову дискретну змінну, яка починається з одиниці, має кількість значень, що дорівнює розміру вектора, і змінюється з одиничним кроком;

- ➡ (3) - ввести i -й елемент вектора та присвоїти йому перше значення;
- ➡ натиснути клавішу , (кома), щоб увести друге значення;
- ➡ (4) - права частина оператора змінить вигляд на таблицю з порожнім полем для введення другого значення вектора;
- ➡ так само ввести третє значення;
- ➡ (5) - вивести елементи вектора у вигляді таблиці значень;
- ➡ (6) - вивести вектор також у стандартному вигляді із круглими дужками.

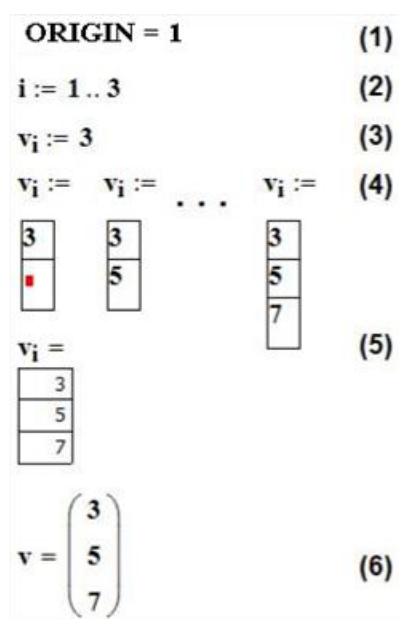


Рис. 1.22. Робота з матрицею за допомогою дискретної змінної

Примітка! Якщо вектор має менше 10 значень, то його виводять у стандартному вигляді із круглими дужками.

	0	1
0	0	
1		

Рис. 1.23. Таблиця даних

Навпаки, коли кількість елементів вектора більше 10, то його виводять у вигляді таблиці значень (Output Table) з номерами рядків і стовпчиків (рис. 1.23).

1.3.2.3. Введення вектора чи матриці за допомогою таблиці даних (Data Table). Виконайте команду головного меню **Додати / Дані / Таблиця** (Insert / Data / Table), і MathCAD вставить порожню таблицю разом з оператором присвоювання (рис. 1.23).

Необхідно ввести ім'я створюваного масиву та необхідні значення елементів масиву. Перехід між комірками таблиці здійснюють клавішами управління курсором. Масив буде мати такий розмір, скільки рядків і стовпчиків були заповнені значеннями [1-3].

1.3.2.4. Створення масиву в результаті виконання вбудованої функції. Вбудована функція **matrix(m, n, f)** повертає матрицю розміром $m \times n$, у якій i, j -й елемент визначений функцією користувача **f(i, j)**. Опис функції **f(i, j)** потрібно розмістити до використання функції **matrix** (рис. 1.24).

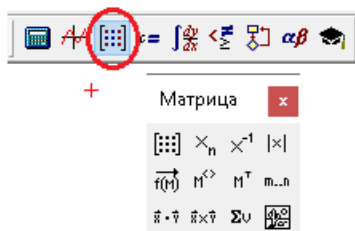
$$a(i,j) := \text{floor}(\text{rnd}(61))$$

$$A := \text{matrix}(2,2,a) = \begin{pmatrix} 17 & 50 \\ 8 & 36 \end{pmatrix}$$

Рис. 1.24. Створення матриці за допомогою вбудованих функцій

1.3.3. Векторні і матричні оператори

Для того щоб почати роботу з векторами і матрицями в середовищі MathCAD, необхідно на панелі математичних інструментів **Математика** активізувати панель векторних і матричних операцій **Матриця** [1-3].



У табл. 1.10 наведені матричні оператори з панелі інструментів **Матриця**.

Таблиця 1.10

Команди панелі векторних і матричних операцій **Матриця**

Кнопка	Клавіші	Назва	Дія
	Ctrl + M	Матриця або вектор	Вставляє в робоче поле порожній шаблон із зазначеною кількістю рядків і стовпців
	[	Нижній індекс	Створює поле для введення нижнього індексу масиву
	-	Обернена матриця	Повертає обернену матрицю для несингулярної квадратної матриці
	Shift + 	Визначник матриці	Знаходить визначник квадратної матриці
	Ctrl + 6	Стовпець матриці	 - повертає j-й стовпчик матриці
	Ctrl + -	Векторизація	Застосовує функцію до вектора та оцінює її символічно. $v := \begin{pmatrix} \frac{3\pi}{2} \\ \frac{\pi}{4} \\ -\frac{\pi}{3} \end{pmatrix} \xrightarrow{\cos(v)} \begin{pmatrix} -1 \\ \frac{\sqrt{2}}{2} \\ \frac{1}{2} \end{pmatrix}$ Наприклад,
	Ctrl + I	Транспонування масиву	Повертає для масиву транспонований масив, помінявши рядки і стовпчики місцями (матриця-рядок стає матрицею-стовпцем, тобто вектором, і навпаки)
	;	Дискретна змінна	Оператор діапазону для створення дискретного виразу
	*	Скалярний добуток	Знаходить скалярний добуток двох векторів
	Ctrl + 8	Векторний добуток	Знаходить векторний добуток двох тривимірних векторів
	Ctrl + 4	Сума елементів вектора	Повертає суму всіх елементів вектора
	Ctrl + T	Картинка	Виділяє область для вставлення картини

Крім доступу до окремих елементів масиву, є можливість здійснювати дії над його підмасивами (наприклад векторами-стовпцями, що утворюють матрицю). Роблять це за допомогою оператора . На рис. 1.25 наведений лістинг виконання деяких операцій.

$$A := \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix} \quad A_{0,0} = 1 \quad A_{2,0} = 7 \quad A_{1,1} = 5$$

$$A^{(0)} = \begin{pmatrix} 1 \\ 4 \\ 7 \end{pmatrix} \quad A^{(1)} = \begin{pmatrix} 2 \\ 5 \\ 8 \end{pmatrix} \quad A^{(2)} = \begin{pmatrix} 3 \\ 6 \\ 9 \end{pmatrix}$$

$$(A^T)^{(0)} = \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix} \quad (A^T)^{(0)T} = (1 \ 2 \ 3)$$

Рис. 1.25. Лістинг деяких операцій із підмасивами

У табл. 1.11 наведені основні операції з векторами та матрицями. Тут кнопки із панелей інструментів **Калькулятор** (Calculator) або **Матриця** (Matrix) [1-3].

Таблиця 1.11

Основні операції з векторами й матрицями

Оператор	Кнопки	Назва	Дія
1	2	3	4
$u \cdot v$		Скалярний добуток векторів однакового розміру	Знаходить суму добутків відповідних елементів векторів
$u \times v$		Векторний добуток векторів розміром 3	Знаходить вектор, перпендикулярний до площини векторів u та v , із модулем $ u $
$v \cdot s, s \cdot v, A \cdot s, s \cdot A$		Добуток масиву на скаляр	Перемножує КОЖНИЙ елемент масиву на скаляр
$\frac{u}{s}, \frac{A}{s}$		Ділення масиву на скаляр	Ділить кожний елемент масиву на скаляр
$v + s, s + v, A + s, s + A$		Сума масиву та скаляра	Додає до кожного елемента масиву скаляр

1	2	3	4
$\mathbf{v} - \mathbf{s},$ $\mathbf{A} - \mathbf{s}$		Різниця масиву та скаляра	Віднімає від кожного елемента масиву скаляр
$-\mathbf{v},$ $-\mathbf{A}$		Зміна знака	Змінює знак кожного елемента масиву на протилежний
$\mathbf{A} \mathbf{u}$		Добуток матриці на вектор	Знаходить добуток матриці з $\mathbf{m}$ стовпчиками на вектор із $\mathbf{m}$ рядками
$\mathbf{A} \mathbf{B}$		Добуток матриці $\mathbf{A}$ на матрицю $\mathbf{B}$ (не завжди дорівнює $\mathbf{B} \mathbf{A}$)	Помножує матрицю з $\mathbf{m}$ стовпчиками на матрицю з $\mathbf{m}$ рядками
$\mathbf{A} + \mathbf{B}, \mathbf{B} + \mathbf{A}$		Сума матриць однакового розміру	Підсумовує відповідні елементи матриць
$\mathbf{A} - \mathbf{B}$		Різниця матриць однакового розміру	Віднімає від елементів $\mathbf{A}$ відповідні елементи $\mathbf{B}$
$\mathbf{M}^n$		Ступінь квадратної матриці	$\mathbf{n} > \mathbf{0}$ – добуток $\mathbf{n}$ матриць; $\mathbf{n} = \mathbf{0}$ – одинична матриця; $\mathbf{n} < \mathbf{0}$ – добуток $\mathbf{n}$ обернених матриць

Приклади реалізації деяких операцій із матрицями наведений на рис. 1.26.

1.3.4. Векторні та матричні функції

Вбудовані функції для роботи з векторами та матрицями можна поділити на три групи:

1. Функції, які повертають скалярні характеристики векторів і матриць (табл. 1.12).
2. Функції, які повертають новий вектор чи матрицю (табл. 1.13).
3. Функції сортування масивів (табл. 1.14).

Деякі операції з матрицями

Вихідні матриці

$$A := \begin{pmatrix} -3 & 2 \\ 4 & 3 \\ 0 & 5 & -9 \end{pmatrix} \quad B := \begin{pmatrix} 1 & 2 & 3 \\ 5 & 6 & 7 \end{pmatrix} \quad C := \begin{pmatrix} 2.3 & 1.2 & 4 \\ 0.1 & 2.5 & 0.6 \\ 2 & 4 & 0 \end{pmatrix} \quad D := \begin{pmatrix} 6 \\ 4 \\ 8 \end{pmatrix} \quad V := \begin{pmatrix} 3 \\ 5 \\ 7 \end{pmatrix}$$

$A_{1,1} = 3$ $\text{ORIGIN} := 1$ $A_{1,1} = -3$

Множення на скаляр

$$M := B \cdot 0.5 \quad M = \begin{pmatrix} 0.5 & 1 & 1.5 \\ 2.5 & 3 & 3.5 \end{pmatrix}$$

Множення матриць

$$L := A \cdot B = \begin{pmatrix} 7 & 6 & 5 \\ 19 & 26 & 33 \\ -20 & -24 & -28 \end{pmatrix}$$

Транспонування матриці

$$L^T = \begin{pmatrix} 7 & 19 & -20 \\ 6 & 26 & -24 \\ 5 & 33 & -28 \end{pmatrix}$$

Обчислення зворотної матриці

$$F := C^{-1} = \begin{pmatrix} 0.107 & -0.712 & 0.413 \\ -0.053 & 0.356 & 0.044 \\ 0.205 & 0.302 & -0.25 \end{pmatrix}$$

Обчислення визначника

$$|C| = -22.48$$

Заміна стовпця на вектор

$$Y := C \quad Y^{(2)} := D \quad Y = \begin{pmatrix} 2.3 & 1.2 & 6 \\ 0.1 & 2.5 & 4 \\ 2 & 4 & 8 \end{pmatrix}$$

Векторний добуток

$$V \cdot D = 94$$

Проведення різних обчислень

$$\beta := 1500 \quad \gamma := 5 \quad m := \begin{pmatrix} 0.25 \\ 0.157 \\ 0.1 \end{pmatrix} \quad N := \sqrt{\frac{2\pi^2 m^4}{16\beta\gamma}} \quad N = \begin{pmatrix} 8.016 \times 10^{-4} \\ 3.161 \times 10^{-4} \\ 1.283 \times 10^{-4} \end{pmatrix}$$

Рис. 1.26. Виконання деяких операцій із матрицями

1.3.4.1. Функції, які повертають скалярні характеристики векторів і матриць, наведені в табл. 1.12 [1-3].

Таблиця 1.12

Функції знаходження скалярних характеристик векторів і матриць

Функція	Результат
cols(A)	Кількість стовпців у матриці <i>A</i>
length(v)	Кількість елементів вектора <i>v</i>
last(v)	Індекс останнього елемента вектора <i>v</i>
max(A)	Найбільший елемент масиву <i>A</i>
min(A)	Найменший елемент масиву <i>A</i>
norm1(M)	L_1 – норма квадратної матриці <i>M</i>
norm2(M)	L_2 – норма квадратної матриці <i>M</i>
norme(M)	Евклідова норма квадратної матриці <i>M</i>
normi(M)	Рівномірна норма квадратної матриці <i>M</i>
rows(A)	Кількість рядків у матриці <i>A</i>
tr(M)	Слід квадратної матриці – сума елементів головної діагоналі
rank(A)	Ранг матриці <i>A</i>

На рис. 1.27 наведені приклади обчислення скалярних характеристик векторів і матриць.

Скалярні характеристики векторів і матриць

Вихідні дані

$$A := \begin{pmatrix} 1 & 4 \\ 2 & 5 \\ 3 & 6 \end{pmatrix} \quad v := \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix} \quad w := (1 \ 2 \ 3)$$

Число рядків у матриці

$$\text{rows}(A) = 3 \quad \text{rows}(v) = 3 \quad \text{rows}(w) = 1$$

Число стовпців у матриці

$$\text{cols}(A) = 2 \quad \text{cols}(v) = 1 \quad \text{cols}(w) = 3$$

Число елементів у векторі $\text{length}(v) = 3$

Індекс останнього елементу вектора $\text{last}(v) = 2$

Рис. 1.27. Обчислення скалярних характеристик векторів і матриць

1.3.4.2. Функції, які повертають новий вектор чи матрицю. Для того щоб зробити з двох або більше матриць одну чи навпаки, у MathCAD передбачено матричні функції, які наведені в табл. 1.13. Приклади їх використання наведені на рис. 1.28 [1-3].

Таблиця 1.13

Функції для створення нових векторів і матриць

Функція	Результат
1	2
augment(s1, s2, s2, ...) augment(v, u, w,...) augment(A, B, C,...)	Масив, утворений приєднанням до масиву <i>A</i> бік у бік праворуч від масиву <i>B</i> (кількість рядків має бути однаковою)
diag(v)	Діагональна квадратна матриця з елементами вектора <i>V</i> на головній діагоналі
diag(M)	Вектор-стовпець із елементів головної діагоналі квадратної матриці
identity(n)	Одинична квадратна матриця розміром $n \times n$ (діагональні елементи = 1, інші елементи = 0)
matrix(m, n, f)	Матриця розміром $m \times n$, у якій i,j -й елемент визначають функцією користувача $f(i, j)$

1	2
stack(s1, s2, s2, ...) stack(v, u, w,...) stack(A, B, C,...)	Масив, утворений приєднанням до масиву A знизу масиву B (кількість стовпців має бути однаковою)
submatrix(A, ir, jr, ic, jc)	Субматриця з елементів матриці A в рядках з i_r по j_r і стовпчиках з i_c по j_c . Якщо $i_r > j_r$, то порядок рядків буде оберненим, якщо $i_c > j_c$, то і порядок стовпчиків буде теж оберненим

Створення матриць на основі існуючих

Вихідні дані

$$A := \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{pmatrix} \quad B := \begin{pmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix}$$

Приєднання до матриці A :
 знизу матриці B праворуч матриці B

$$\text{stack}(A, B) = \begin{pmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{pmatrix} \quad C := \text{augment}(A, B) = \begin{pmatrix} 1 & 2 & 3 & 0 & 0 & 0 \\ 4 & 5 & 6 & 0 & 0 & 0 \end{pmatrix}$$

Вставка стовпців у матрицю

$$\text{augment}\left[\begin{pmatrix} 9 \\ 9 \end{pmatrix}, C, \begin{pmatrix} 7 \\ 7 \end{pmatrix}\right] = \begin{pmatrix} 9 & 1 & 2 & 3 & 0 & 0 & 0 & 7 \\ 9 & 4 & 5 & 6 & 0 & 0 & 0 & 7 \end{pmatrix}$$

$$\text{augment}\left[\begin{pmatrix} 9 \\ 9 \end{pmatrix}, A, \begin{pmatrix} 8 \\ 8 \end{pmatrix}, B, \begin{pmatrix} 7 \\ 7 \end{pmatrix}\right] = \begin{pmatrix} 9 & 1 & 2 & 3 & 8 & 0 & 0 & 0 & 7 \\ 9 & 4 & 5 & 6 & 8 & 0 & 0 & 0 & 7 \end{pmatrix}$$

$$M := \begin{pmatrix} 1 & 7 & 1 & 4 & 4 \\ -5 & -8 & -2 & 3 & 3 \\ -6 & -9 & -3 & 2 & 3 \\ 1 & 2 & 3 & 4 & 3 \\ 4 & 5 & 5 & 6 & 8 \end{pmatrix}$$

Вибір елементів матриці, які розміщені між рядками 1 і 2 та стовпцями 0 і 2 (включно)

$$\text{submatrix}(M, 1, 2, 0, 2) = \begin{pmatrix} -5 & -8 & -2 \\ -6 & -9 & -3 \end{pmatrix}$$

Перестановка двох останніх аргументів змінює порядок стовпців

$$\text{submatrix}(M, 1, 2, 2, 0) = \begin{pmatrix} -2 & -8 & -5 \\ -3 & -9 & -6 \end{pmatrix}$$

Перестановка двох перших скалярних аргументів змінює порядок рядків

$$\text{submatrix}(M, 2, 1, 2, 0) = \begin{pmatrix} -3 & -9 & -6 \\ -2 & -8 & -5 \end{pmatrix}$$

Рис. 1.28. Створення матриць із існуючих

Існує ряд матриць спеціального виду (верхні і нижні трикутні, одиничні, скалярні, нульові і т. д.). У MathCAD є декілька функцій, які дозволяють швидко і просто задавати матриці такого типу. Для того щоб ввести таку функцію, достатньо просто набрати її з клавіатури або через меню **Додати** → **Функцію** → знайти в **Категорії функцій** розділ **Вектори і матриці** і вибрати потрібну функцію.

На рис. 1.29 наведені приклади створення матриць спеціального виду.

Функції задання матриць спеціального виду

$$N := \text{identity}(3) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad V := \begin{pmatrix} 5 \\ 7 \\ 9 \end{pmatrix} \quad W := \text{diag}(V) = \begin{pmatrix} 5 & 0 & 0 \\ 0 & 7 & 0 \\ 0 & 0 & 9 \end{pmatrix}$$

Обчислення рангу матриці

$$\text{rank}(N) = 3 \quad \text{rank}(V) = 1 \quad \text{rank}(W) = 3$$

Властивості одиничної та оберненої матриць

$$M := \begin{pmatrix} 4 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix} \quad |M| = -9 \quad E := \text{identity}(3) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

$$M^{-1} \cdot M = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = M \cdot M^{-1} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} = E$$

$$E \cdot M = \begin{pmatrix} 4 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix} = M \cdot E = \begin{pmatrix} 4 & 2 & 3 \\ 4 & 5 & 6 \\ 7 & 8 & 9 \end{pmatrix} = M$$

Рис. 1.29. Створення матриць спеціального виду

1.3.4.3. Функції сортування масивів. Починаючи із третьої версії, у системі MathCAD з'явилися функції сортування – перестановки елементів векторів і матриць (табл.1.14).

Функції сортування елементів векторів і матриць

Функція	Результат
sort(v)	Упорядковує елементи вектора V за зростанням
csort(A, n)	Упорядковує елементи A в стовпчику номер n за зростанням перестановкою рядків
rsort(A, n)	Упорядковує елементи A в рядку номер n за зростанням перестановкою стовпчиків
reverse (v)	Обертає порядок елементів вектора V
reverse (sort(v))	Упорядковує елементи вектора V за спаданням
reverse (A)	Обертає порядок рядків матриці A

На рис. 1.30 наведені приклади сортування елементів векторів і матриць.

Сортування елементів векторів і матриць

$$\begin{aligned}
 \mathbf{v} &:= \begin{pmatrix} 5 \\ 1 \\ 2 \end{pmatrix} & \mathbf{sort}(\mathbf{v}) &= \begin{pmatrix} 1 \\ 2 \\ 5 \end{pmatrix} & \mathbf{reverse}(\mathbf{v}) &= \begin{pmatrix} 2 \\ 1 \\ 5 \end{pmatrix} & \mathbf{reverse}(\mathbf{sort}(\mathbf{v})) &= \begin{pmatrix} 5 \\ 2 \\ 1 \end{pmatrix} \\
 \mathbf{B} &:= \begin{pmatrix} 5 & 3 & 8 \\ 1 & 0 & 6 \\ 2 & 4 & 9 \end{pmatrix} & \mathbf{csort}(\mathbf{B}, 1) &= \begin{pmatrix} 1 & 0 & 6 \\ 5 & 3 & 8 \\ 2 & 4 & 9 \end{pmatrix} & \mathbf{reverse}(\mathbf{B}) &= \begin{pmatrix} 2 & 4 & 9 \\ 1 & 0 & 6 \\ 5 & 3 & 8 \end{pmatrix}
 \end{aligned}$$

Рис. 1.30. Сортування елементів векторів і матриць

ЗАВДАННЯ ДЛЯ ВИКОНАННЯ

Виконати свій варіант, заданий викладачем, застосовуючи матричні та векторні оператори і функції. Якщо потрібно, то задати початкові дані матриць самостійно (табл. 1.15).

Таблиця 1.15

Варі- ант	Зміст
1	2
1	Задана квадратна матриця A та одинична матриця E порядку n = 4. 1. Сформувати вектор v з елементів третього стовпчика матриці A. 2. Сформувати діагональну матрицю B з елементів вектора v. 3. Обчислити ранг і слід матриці A. 4. Обчислити $C = (A - E) A^T$, де A^T – транспонована матриця A. 5. Обчислити другу норму матриць A та C. 6. Сформувати нову матрицю D, розташувавши матрицю A^T над матрицею A. 7. Знайти min і max елементи матриць A, C і вектора v. 8. Упорядкувати елементи вектора v за зростанням і спаданням. 9. Обчислити елементи вектора $w = 0,5 (A + E) v$. 10. Обчислити s – суму елементів вектора v. 11. Обчислити скалярний добуток векторів v і w. 12. Знайти довжину вектора w
2	Задана квадратна матриця A та одинична матриця E порядку n = 5. 1. Сформувати вектор v з елементів першого рядка матриці A. 2. Сформувати нову матрицю D, вилучивши з матриці A, елементи розташовані з першого по третій стовпець, і рядки із другого по третій. 3. Визначити кількість рядків і стовпців матриці D. 4. Сформувати матрицю $B = A^T$. 5. Обчислити $w = (A + A^3 - E) v$. 6. Обчислити p – довжину вектора w. 7. Обчислити $c = (A + E + A^2) p w$. 8. Сформувати нову матрицю F, приєднавши до матриці A матрицю B бік у бік. 9. Обчислити слід та Евклідову норму матриці A. 10. Упорядкувати елементи вектора c за зростанням і спаданням. 11. Знайти min і max елементи матриць A та D. 12. Обчислити детермінант матриці B
3	Задана квадратна матриця A та одинична матриця E порядку n = 5. 1. Сформувати вектор v з елементів другого стовпчика матриці A. 2. Обчислити вектор $w = (A - E) v$. 3. Обчислити p – довжину вектора w. 4. Обчислити s – суму елементів вектора v. 5. Обчислити скалярний добуток векторів v та w. 6. Сформувати матрицю $B = A^T$.

1	2
	7. Обчислити першу норму матриці B. 8. Сформувати матрицю D, вилучивши з матриці A рядки з першого по третій і стовпці із другого по третій. 9. Сформувати нову матрицю F, розташувавши матрицю A над матрицею B. 10. Визначити кількість рядків і стовпців матриці F. 11. Упорядкувати елементи вектора v за зростанням і спаданням. 12. Обчислити слід матриці B
4	Задана квадратна матриця A та одинична матриця E порядку n = 5. 1. Сформувати вектор v з елементів останнього стовпця матриці A. 2. Обчислити вектор $\mathbf{w} = (\mathbf{A} + \mathbf{A}^2 - \mathbf{E}) \mathbf{v}$. 3. Обчислити p – довжину вектора v. 4. Обчислити s – суму елементів вектора w. 5. Обчислити $\mathbf{C} = \mathbf{p} \mathbf{A} - \mathbf{s} \mathbf{E}$. 6. Обчислити слід матриці A. 7. Обчислити ранг матриці A. 8. Обчислити рівномірну норму матриці A. 9. Сформувати матрицю B, транспонувавши матрицю A. 10. Сформувати нову матрицю D, розташувавши матрицю A над матрицею B. 11. Визначити кількість рядків і стовпців матриці D. 12. Упорядкувати елементи другого стовпця матриці A за зростанням
5	Задана квадратна матриця A та одинична матриця E порядку n = 5. 1. Сформувати вектор v з елементів третього стовпця матриці A. 2. Сформувати матрицю B, транспонувавши матрицю A: $\mathbf{B} = \mathbf{A}^T$. 3. Сформувати матрицю C, вилучивши з матриці B другий і третій стовпці. 4. Сформувати матрицю D, приєднавши до матриці A матрицю B бік у бік. 5. Визначити кількість рядків і стовпців у матриці D. 6. Обчислити l – довжину вектора v. 7. Обчислити $\mathbf{C} = \mathbf{l} (\mathbf{A} + \mathbf{A}^2 + \mathbf{E})$. 8. Обчислити слід матриці C. 9. Обчислити Евклідову норму матриці A. 10. Упорядкувати елементи вектора v за зростанням і спаданням. 11. Обчислити детермінант матриці C. 12. Знайти min і max елементи матриць A, B, C і вектора v

1	2
6	Задана квадратна матриця A та одинична матриця E порядку n = 4. 1. Сформувати вектор v з елементів третього стовпця матриці A. 2. Обчислити p – довжину вектора v. 3. Обчислити матрицю D = Aⁿ. 4. Обчислити C = (A - E) A^T. 5. Сформувати матрицю F, вилучивши з матриці A перший стовпець. 6. Обчислити слід матриці A. 7. Визначити кількість стовпців і рядків матриці F. 8. Упорядкувати елементи першого рядка матриці A за зростанням. 9. Знайти min і max елементи матриць A, F, D. 10. Обчислити суму елементів вектора v. 11. Обчислити S = A⁻¹. 12. Обчислити w = p v
7	Задана квадратна матриця A та одинична матриця E порядку n = 5. 1. Сформувати матрицю B = A^T, де A^T – транспонована матриця A. 2. Сформувати вектор v з елементів четвертого рядка матриці A. 3. Обчислити слід матриці B. 4. Обчислити другу норму матриці A. 5. Обчислити d - детермінант матриці A. 6. Обчислити C = (d E + 1/ d A) B. 7. Сформувати матрицю D, вилучивши з матриці A третій стовпець. 8. Знайти min і max значення елементів вектора v і матриць A, B, C, D. 9. Упорядкувати елементи вектора v за зростанням і спаданням. 10. Знайти довжину вектора v. 11. Обернути порядок рядків матриці A. 12. Визначити кількість рядків і стовпців матриці C
8	Задана квадратна матриця A та одинична матриця E порядку n = 5. 1. Сформувати вектор v з елементів другого стовпця матриці A. 2. Упорядкувати елементи вектора v за зростанням. 3. Сформувати матрицю C = A^T, де A^T – транспонована матриця A. 4. Сформувати матрицю B, розташувавши матрицю C над матрицею A. 5. Визначити другу норму матриці A. 6. Сформувати діагональну матрицю D з елементів вектора v. 7. Упорядкувати матрицю A за зростанням елементів її першого стовпчика. 8. Обчислити p – детермінант матриці A. 9. Обчислити F = E + A + A². 10. Одержати вектор w = A v. 11. Упорядкувати елементи вектора w за спаданням. 12. Знайти ранг матриці C

1	2
9	Задані квадратні матриці A і B та одинична матриця E порядку n = 5. 1. Сформувати вектор v з елементів другого стовпця матриці A. 2. Сформувати вектор w за формулою $\mathbf{A A}^T \mathbf{B v}$, де $\mathbf{A}^T$ – транспонована матриця A. 3. Знайти max і min значення матриць A, B, $\mathbf{A}^T$ і векторів w, v. 4. Сформувати матрицю G, приєднавши до матриці A матрицю B бік у бік. 5. Знайти число m, що відповідає кількості стовпчиків матриці G. 6. Обчислити детермінант матриці A. 7. Обчислити довжину вектора w. 8. Обчислити $\mathbf{A}^2$. 9. Знайти добуток матриці A на вектор v. 10. Знайти суму двох матриць (A + B). 11. Обчислити $\mathbf{C} = \mathbf{A A}^T \mathbf{E}$. Обчислити першу норму матриці C. 12. Сформувати матрицю H, упорядкувавши третій рядок матриці C за зростанням
10	Задана квадратна матриця A та одинична матриця E порядку n = 5. 1. Сформувати вектор v з елементів другого стовпця матриці A. 2. Визначити кількість рядків матриці A. 3. Сформувати матрицю $\mathbf{C} = \mathbf{A}^T$. 4. Сформувати вектор u з елементів другого рядка матриці C. 5. Знайти скалярний добуток u v. 6. Сформувати вектор $\mathbf{w} = \mathbf{A v}$ як добуток матриці A на вектор v. 7. Обчислити $\mathbf{d} = (\mathbf{A} - \mathbf{E}) \mathbf{w}$. 8. Обчислити першу норму матриці C. 9. Обчислити обернену матрицю $\mathbf{A}^{-1}$. 10. Визначити останній індекс вектора v. 11. Обчислити слід матриці C. 12. Знайти суму елементів вектора w

1.4. Робота з масивами, базові операції розв’язання СЛАР

Визначення методів обчислення елементів матриць із використанням можливостей системи MathCAD; знаходження розв’язків системи лінійних рівнянь (СЛАР).

1.4.1. Масиви в MathCAD і робота з ними

Масив – сукупність кінцевої кількості числових чи символьних елементів, що має унікальне ім'я, які впорядковані певним чином і мають певні адреси.

У MathCAD умовно виділяють два типи масивів:

- вектори (одноіндексні масиви), матриці (двоіндексні) і тензори (багатоіндексні);
- ранжовані змінні (range variables) – вектори, елементи яких певним чином залежать від їхнього індексу.

Ми будемо використовувати масиви двох типів (табл. 1.16).

Таблиця 1.16

Одновимірний масив (<i>вектор</i>)	Двовимірний масив (<i>матриця</i>)
$a := \begin{pmatrix} 15 \\ 1.2 \\ -3 \end{pmatrix} \quad \begin{array}{l} a_0 = 15 \\ a_1 = 1.2 \\ a_2 = -3 \end{array}$	$A := \begin{pmatrix} 1 & 2 & 3 \\ 0.1 & 0.2 & 0.3 \\ 4 & -5 & -2.1 \end{pmatrix}$ $A_{0,0} = 1$ $A_{2,1} = -5$

Нумерація елементів у масиві, як і в матриці, здійснювана з **0**. Для зміни нумерації використовують змінну **ORIGIN** [1-3].

Правила роботи з масивами такі, як і для роботи з матрицями, і визначені в теоретичних відомостях для практичного завдання 2.

1.4.2. Визначення коренів алгебраїчного рівняння $f(x) = 0$

У системі MathCAD рівняння і системи рівнянь можна розв'язувати чисельними та аналітичними методами.

Багато рівнянь, наприклад, трансцендентні, не мають аналітичного розв'язку. Однак їх можна розв'язувати чисельними методами із заданою

похибкою, яка визначена системною змінною **TOL** (за принципом замовчення **TOL** = 10^{-3} , для зміни потрібно задати інше значення, наприклад **TOL** := 10^{-7}).

1.4.2.1. Використання функції root. Для розв'язання одного рівняння з одним невідомим вигляду $f(x) = 0$ використовують функцію **root**. Аргументами цієї функції є вираз і змінна, що входить у вираз. Функція повертає значення змінної, за якої вираз перетворюється на нуль [1-3].

root (f(x), x) повертає значення x, за якого вираз чи функція $f(x)$ перетворюється в 0. Обидва аргументи цієї функції мають бути скалярами. Функція повертає скаляр.

Перший аргумент є або функцією, визначеною де-небудь у робочому документі, або виразом. Вираз має повертати скалярні значення.

Другий аргумент – ім'я змінної, яке використано у виразі. Це та змінна, варіюючи яку, MathCAD намагатиметься перетворити вираз у нуль. Цій змінній перед використанням функції **root** необхідно присвоїти числове значення. MathCAD використовує його як початкове наближення для пошуку кореня.

Приклад 1. Розв'язати рівняння $e^x = x^3$.

Завдання розв'язання рівняння вигляду $f(x) = g(x)$ еквівалентне задачі пошуку кореня виразу $f(x) - g(x) = 0$. Для цього функція **root** може бути використана в такий спосіб: **root (f(x) - g(x), x)** (рис. 1.31).

$$\begin{array}{l} x := 3 \\ \text{root}(e^x - x^3, x) = 1.857 \end{array}$$

Рис. 1.31. Лістинг прикладу 1

Коли рівняння має кілька розв'язків, тоді вибір розв'язку визначений вибором початкового значення змінної. Перше застосування цієї функції допомагає знайти перший корінь **X1**. Для пошуку другого кореня **X2** перший

виключено діленням $f(x)$ на $(x-X1)$. Відповідно, для пошуку третього кореня $X3$ $f(x)$ ділять ще й на $(x-X2)$.

Приклад 2. Розв'язати рівняння $f(x) = 2x^3 - 5x^2 + 43x - 92 = 0$.

Розв'язок показано на рис. 1.32.

```

a3 := 2  a2 := -5  a1 := 43  a0 := -92      Коефіцієнти
F(x) := a3·x3 + a2·x2 + a1·x + a0          Задання функції
      Обчислення дійсного кореню
x := 2    x1 := root(F(x),x)                  x1 = 2.12
      Обчислення інших коренів
i := √-1    x := -1 + 1·i    x2 := root( (F(x)/(x-x1)),x )    x2 = 0.19 + 4.654i
x3 := root( [ F(x)/(x-x1)·(x-x2) ],x )          x3 = 0.19 - 4.654i

```

Рис. 1.32. Лістинг прикладу 2

Для знаходження коренів виразу, що має вигляд

$$a_n x^n + a_{n-1} x^{n-1} + \dots + a_2 x^2 + a_1 x + a_0,$$

краще використовувати функцію **polyroots**, ніж **root**. На відміну від функції **root**, функція **polyroots** не потребує початкового наближення і повертає відразу всі корені, як дійсні, так і комплексні [1-3].

1.4.2.2. Використання функції polyroots. **Polyroots(v)** – повертає корінь полінома ступеня **n**. Коефіцієнти полінома знаходяться у векторі **V** завдовжки **n+1**. Вектор **v** зручно створювати командою **(Symbolics => Polynomial coefficients)**.

Також такі рівняння можна розв'язувати засобами символної математики.

Приклад 3. Знайти розв'язок рівняння за допомогою функції **polyroots**

$$f(x) = 41x^4 + 25x^3 - 33x^2 + 8x - 12 = 0.$$

Алгоритм виконання:

- ➡ записати многочлен на робочий аркуш $41x^4 + 25x^3 - 33x^2 + 8x - 12$;
 - ➡ виділити змінну x в будь-якому місці виразу і викликати команду
- з меню **Symbolic => Polynomial Coefficients** (Symbolic => PolynomialCoefficients) (рис. 1.33);

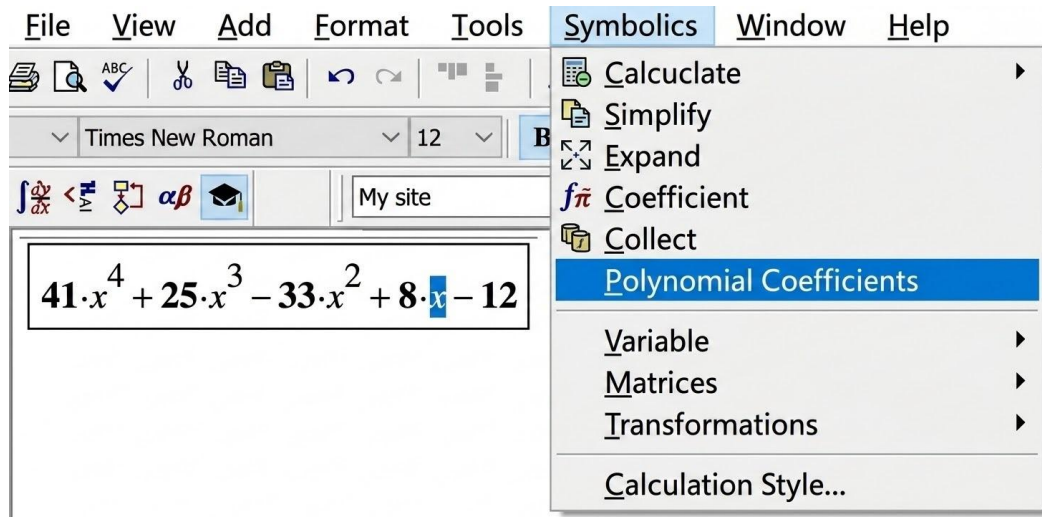


Рис. 1.33

- ➡ за допомогою цієї команди буде створений вектор коефіцієнтів
- многочлена

$$41x^4 + 25x^3 - 33x^2 + 8x - 12$$

$$\begin{pmatrix} -12 \\ 8 \\ -33 \\ 25 \\ 41 \end{pmatrix}$$

$$V := \begin{pmatrix} -12 \\ 8 \\ -33 \\ 25 \\ 41 \end{pmatrix};$$

- ➡ змінній V присвоїти її значення
- ➡ викликати вбудовану функцію знаходження коренів **poliroots** (V).

Результат показано на рис. 1.34.

$$41x^4 + 25x^3 - 33x^2 + 8x - 12 \quad V := \begin{pmatrix} -12 \\ 8 \\ -33 \\ 25 \\ 41 \end{pmatrix} \quad \text{polyroots}(V) = \begin{pmatrix} -1.395 \\ 0.01 + 0.524i \\ 0.01 - 0.524i \\ 0.764 \end{pmatrix}$$

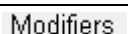
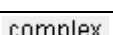
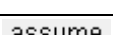
Рис. 1.34. Приклад використання функції **Polyroots**

1.4.2.3. Використання символьної математики. У меню **Символіка** (Symbolics) передбачено низку операцій, орієнтованих на виділену змінну, що використана у виразі. Наприклад, команда **Solve** (Вирішити) обчислює корінь функції, заданої цим виразом.

У табл. 1.17 наведені операції панелі **Символьна** (Symbolic) і клавіші, за якими вони закріплені.

Таблиця 1.17

Операції панелі (Symbolic)

Пікто- грама	Функція	Приклад	Кла- віші
1	2	3	4
	Символьне обчислення	$\frac{d^3}{dt^3} t^5 \rightarrow 60 \cdot t^2$	Ctrl + .
	Символьне обчислення з ключовим словом	$x^2 + 2x + 1 \text{ factor} \rightarrow (x + 1)^2$	Ctrl + >
	Додаткові модифікатори		
	Числове обчислення	$\ln(2) \cdot x^2 \text{ float}, 3 \rightarrow .693 \cdot x^2$	
	Комплексне обчислення	$\frac{2}{3 + 5i} \text{ complex} \rightarrow \frac{3}{17} - \frac{5}{17} \cdot i$	
	Символьне обчислення за деяких обмежень	$\sqrt{M^2} \left \begin{array}{l} \text{simplify} \\ \text{assume}, M < 0 \end{array} \right. \rightarrow -M$	

1	2	3	4
solve	Розв'язок систем і рівнянь	$\begin{pmatrix} x + y = 2 \\ 2x - y = 1 \end{pmatrix} \text{ solve, } \begin{pmatrix} x \\ y \end{pmatrix} \rightarrow \begin{pmatrix} 1 \\ 1 \end{pmatrix}$	
simplify	Спрощення виразів	$\sin(x)^2 + \cos(x)^2 \text{ simplify} \rightarrow 1$	
substitute	Заміна змінної виразом	$\sqrt{1 - x^2} \text{ substitute , } x = 3 \rightarrow (-8)^{\left(\frac{1}{2}\right)}$	
factor	Розклад на множники	$x^3 + x \text{ factor} \rightarrow x \cdot (x^2 + 1)$	
expand	Перемноження степенів і добутків	$\frac{x^2 + 1}{x} \text{ expand} \rightarrow x + \frac{1}{x}$	
coeffs	Визначення коефіцієнтів полінома	$4x - 5 \text{ coeffs, } x \rightarrow \begin{pmatrix} -5 \\ 4 \end{pmatrix}$	
collect	Групування за степенями змінної	$(3x + y)^2 \text{ collect , } y \rightarrow y^2 + 6 \cdot x \cdot y + 9 \cdot x^2$	
series	Розклад у ряд Тейлора чи Лорана	$e^x \text{ series , } x \rightarrow 1 + x + \frac{1}{2} \cdot x^2 + \frac{1}{6} \cdot x^3 + \frac{1}{24} \cdot x^4 + \frac{1}{120} \cdot x^5$	
parfrac	Розклад на елементарні дроби	$\frac{2 - x}{x^3 + x} \text{ convert , parfrac, } x \rightarrow \frac{2}{x} - \frac{(1 + 2 \cdot x)}{(x^2 + 1)}$	
m ^T →	Транспонування матриці	$A := \begin{pmatrix} 1 & 2 \\ 3 & 4 \end{pmatrix} \quad A^T \rightarrow \begin{pmatrix} 1 & 3 \\ 2 & 4 \end{pmatrix}$	
m ⁻¹ →	Відшукування оберненої матриці	$A^{-1} \rightarrow \begin{pmatrix} -2 & 1 \\ \frac{3}{2} & \frac{-1}{2} \end{pmatrix}$	
m →	Знаходження визначника матриці	$ A \rightarrow -2$	

Використання символьної математики розглянемо на прикладах.

Приклад 4. Знайти розв'язок кубічного рівняння з використанням символьної математики.

$$f(x) = x^3 + 3x^2 - 972 = 0.$$

Алгоритм виконання:

- ➡ записати многочлен на робочий аркуш $x^3 + 3x^2 - 972$;
- ➡ виділити змінну x в будь-якому місці виразу;
- ➡ викликати команду із меню **Символіка** => **Змінні** => **Обчислити** (Symbolic => Variable => Solve).

Результат показано на рис. 1.35.

$$x^3 + 3x^2 - 972$$

$$\left(\begin{array}{c} 9 \\ -6 + 6i\sqrt{2} \\ -6 - 6i\sqrt{2} \end{array} \right)$$

Рис. 1.35. Приклад використання символічної математики

Приклад 5. В аналітичній формі отримати всі корені полінома другого степеня: спочатку із застосуванням команди **solve** для розв'язання, а потім **simplify** для спрощення результату (рис. 1.36).

$$a \cdot x^2 + b \cdot x + c \text{ solve, } x \rightarrow \left(\begin{array}{c} \frac{\frac{b}{2} + \frac{\sqrt{b^2 - 4 \cdot a \cdot c}}{2}}{a} \\ \frac{\frac{b}{2} - \frac{\sqrt{b^2 - 4 \cdot a \cdot c}}{2}}{a} \end{array} \right) \text{ simplify } \rightarrow \left(\begin{array}{c} \frac{b + \sqrt{b^2 - 4 \cdot a \cdot c}}{2 \cdot a} \\ \frac{b - \sqrt{b^2 - 4 \cdot a \cdot c}}{2 \cdot a} \end{array} \right)$$

Рис. 1.36. Аналітична форма розв'язання полінома другого степеня

1.4.3. Розв'язання систем лінійних рівнянь

Розв'язок системи лінійних рівнянь у матричній формі $A \cdot X = B$ визначений співвідношенням

$$X = A^{-1} \cdot B;$$

B – вектор вільних членів.

1.4.3.1. Розв’язання матричних рівнянь. Розглянемо систему n лінійних алгебраїчних рівнянь відносно n невідомих $x_1, x_2, \dots, x_n$:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1, \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = b_2, \\ \cdot \quad \cdot \quad \cdot \quad \cdot \quad \cdot \quad \cdot \quad \cdot \quad \cdot \\ a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nn}x_n = b_n. \end{cases} \quad (1.1)$$

$$\mathbf{Ax} = \mathbf{b}, \quad (1.2)$$

$$\text{де } A = \begin{vmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{vmatrix}, \quad x = \begin{vmatrix} x_1 \\ x_2 \\ \dots \\ x_n \end{vmatrix}, \quad b = \begin{vmatrix} b_1 \\ b_2 \\ \dots \\ b_n \end{vmatrix}. \quad (1.3)$$

Якщо матриця A – не вироджена, тобто $\det A \neq 0$, то система (1.1) чи еквівалентне їй матричне рівняння (1.2) має єдиний розв’язок.

Справді, за умови $\det A \neq 0$ існує обернена матриця A^{-1} .
Перемножуючи обидві частини рівняння (1.2) на матрицю A^{-1} , одержимо

$$A^{-1} Ax = A^{-1} b,$$

$$x = A^{-1} b. \quad (1.4)$$

Формула (1.4) дає розв'язок рівняння (1.2), і він єдиний.

1.4.3.2. Розв'язок матричних рівнянь за допомогою функції *lsolve*.

Системи лінійних рівнянь зручно розв'язувати за допомогою функції *lsolve*.

lsolve(A, b) – повертає вектор-розв'язок x такий, що $Ax = b$.

Аргументи функції:

A – квадратна, не сингулярна матриця.

b – вектор, що має стільки ж рядів, скільки рядів у матриці A .

На рис. 1.37 показаний приклад розв'язання системи трьох лінійних рівнянь відносно трьох невідомих матричним способом і за допомогою функції *lsolve* [1-3].

Mathcad - [Untitled:1]

File Edit View Insert Format Tools Symbolics Window Help

Система рівнянь	Матриця системи	Вектор вільних членів
$\begin{cases} 3x_1 - x_2 = 5 \\ -2x_1 + x_2 + x_3 = 0 \\ 2x_1 - x_2 + 4x_3 = 15 \end{cases}$	$A := \begin{pmatrix} 3 & -1 & 0 \\ -2 & 1 & 1 \\ 2 & -1 & 4 \end{pmatrix}$	$b := \begin{pmatrix} 5 \\ 0 \\ 15 \end{pmatrix}$

Обчислення детермінанту
 $|A| = 5$ Відмінний від нуля, система має один розв'язок

Розв'язання системи
 матричним способом за допомогою функції *lsolve*

$x := A^{-1} \cdot b$	$x := \begin{pmatrix} 2 \\ 1 \\ 3 \end{pmatrix}$	$x := \text{lsolve}(A, b) = \begin{pmatrix} 2 \\ 1 \\ 3 \end{pmatrix}$
-----------------------	--	--

Перевірка $A \cdot x - b = \begin{pmatrix} 0 \\ 0 \\ 0 \end{pmatrix}$

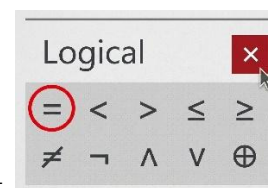
Рис. 1.37. Розв'язання матричних рівнянь

1.4.3.3. Розв'язання матричних рівнянь за допомогою функції *Isolve*

Розв'язання систем рівнянь за допомогою блока **Given – Find**

Обчислювальний блок **Given-Find** хороший тим, що він здатний надати результат розв'язання системи рівнянь у двох виглядах: *чисельному* і *символьному*. Розглянемо кожен із них детальніше [1-3].

Чисельний метод застосовують у тому випадку, коли необхідно отримати тільки чисельні результати своєї роботи. У цьому випадку необхідно спочатку задати значення всіх «літер» і *змінних (!)*. Змінні потрібно задати як початкові наближення. Ці початкові наближення потрібні для коректної роботи чисельних методів MathCAD. При цьому якщо початкове наближення не задано або задано неправильно, то висока ймовірність, що розв'язок знайдено не буде.



Далі через «жирне» дорівнює (із панелі **Логічний** –) записують список рівнянь системи, а після нього **Find (x, y, z,...) →** або **"=**".

Символьний метод застосовують для знаходження виразу шуканої змінної із системи. У цьому випадку не обов'язково задавати всі величини, які входять у систему. Досить просто записати всі рівняння в порядку і потім знайти розв'язок у вигляді виразів за допомогою оператора **Find (x, y, z,...) →**. Слід зазначити, що не завжди вдається отримати символьний вираз для змінної, зважаючи на складність перетворень [1-3].

Для розв'язання систем нелінійних рівнянь організовано спеціальний обчислювальний блок, який починається зі службового слова **Given** і має таку структуру:

початкові наближення

Given

рівняння

обмеження

оператори з функціями **find** або **minerr**

Початкові умови визначають початкові значення змінних, які обчислюють, і задані за допомогою оператора присвоювання.

Рівняння записують із застосуванням жирного знаку дорівнює (**Ctrl** + =) між лівою і правою частинами рівнянь.

Обмеження задають як нерівності або рівності, які мають бути виконані для розв'язання системи рівнянь. У деяких задачах обмеження можуть бути відсутні. Для формування обмежень використовують логічні операції >, <, ≥, ≤, ≠, =.

У блоці може бути використана одна з таких функцій:

- **Find (v1,..., vn)** – повертає значення однієї або ряду змінних для точного розв'язання;

- **Minerr (v1,..., vn)** – повертає значення однієї або ряду змінних для наближеного розв'язання.

Між цими функціями існують принципові відмінності. Першу функцію використовують тоді, коли розв'язок реально існує (хоча і не є аналітичним). Друга функція намагається знайти максимальне наближення навіть до неіснуючого розв'язку через мінімізацію середньоквадратичної похибки рішення [1-3].

Функції **find** і **minerr** можна використовувати для розв'язання одного рівняння або системи. Кількість рівнянь має дорівнювати кількості невідомих. Аргументи **v1, v2,...,vn** – це скалярні змінні, значення яких шукають у блоці обчислень. Коли блок розв'язання рівнянь шукає одну невідому, функції повертають скаляр. Якщо невідомих декілька, то повертають вектор, першим елементом якого є шукане значення змінної **v1**, другим елементом – **v2** і т. д.

Приклад. Розв'язати систему лінійних рівнянь
$$\begin{cases} 3x + 4y - 5z = 1 \\ 7x - 4y + z = -2 \\ 2y - 9z = 5 \end{cases}.$$

Розв'язання показано на рис. 1.38.

$\begin{array}{l} x := 0 \quad y := 0 \quad z := 0 \\ \textbf{Given} \\ 3 \cdot x + 4 \cdot y - 5 \cdot z = 1 \\ 7 \cdot x - 4 \cdot y + z = -2 \\ 2 \cdot x - 9z = 5 \end{array}$ $\textbf{Find}(x, y, z) \rightarrow \begin{pmatrix} -\frac{29}{82} \\ \frac{91}{328} \\ -\frac{26}{41} \end{pmatrix} = \begin{pmatrix} -0.354 \\ -0.277 \\ -0.634 \end{pmatrix}$ а	$\begin{array}{l} x := 0 \quad y := 0 \quad z := 0 \\ \textbf{Given} \\ 3 \cdot x + 4 \cdot y - 5 \cdot z = 1 \\ 7 \cdot x - 4 \cdot y + z = -2 \\ 2 \cdot x - 9z = 5 \end{array}$ $\textbf{Minerr}(x, y, z) \rightarrow \begin{pmatrix} -\frac{29}{82} \\ \frac{91}{328} \\ -\frac{26}{41} \end{pmatrix} = \begin{pmatrix} -0.354 \\ -0.277 \\ -0.634 \end{pmatrix}$ б
---	---

Рис. 1.38. Розв'язання систем рівнянь за допомогою блоків
Given – Find (а) і Given – Minerr (б)

ЗАВДАННЯ ДЛЯ ВИКОНАННЯ

Завдання 1. Задати будь-які два вектори, розмірністю 3, що складаються з двозначних чисел: додатних і від'ємних;

1. Виконати такі операції:

- обчислити номер останнього елемента кожного вектора;
- обчислити суму компонентів кожного вектора;
- відсортувати перший вектор за зростанням, другий – за зменшенням;
- транспонувати отримані вектори;
- обчислити (векторний і скалярний) добуток векторів, знайти їхню суму і різницю.

2. Задати вектор $\mathbf{v}$ (розмірність не менше 3, значення довільні), сформувати вектор $\mathbf{w}$, елементи якого задовольняють умову $w_i = \lg(2v_i)^2$.

(Використати операцію векторизації.) В отриманому векторі переставити елементи вектора у зворотному порядку.

Завдання 2. Створити матрицю M розмірністю 9×9 , використовуючи функції склеювання і стикування (1.2) і вектори із завдання 1. Встановити індекси елементів матриці, починаючи з номера прізвища за списком в журналі.

1. Визначити:

- слід матриці;
- максимальний і мінімальний елементи матриці;
- обчислити визначник матриці;
- відсортування матрицю за третім рядком;
- вивести на екран перший стовпець і поелементно другий рядок.

2. Вирізати матрицю 4×4 із верхніх лівих елементів створеної матриці.

3. Транспонувати цю матрицю. Обчислити її визначник і слід.

Завдання 3. Розв'язати рівняння та зробити перевірку (варіанти в табл. 1.18 і ще один на вибір), використовуючи **root** і **polyroots**.

Таблиця 1.18

Варіанти для завдання 3

Варіант	Завдання	Варіант	Завдання
1	$f(x) = x^3 - 6x^2 + 9x + 2$	2	$f(x) = x^3 - 12x^2 + 45x - 51$
3	$f(x) = -x^3 + 18x^2 - 105x + 193$	4	$f(x) = x^3 - 9x^2 + 24x - 15$
5	$f(x) = -x^3 - 12x^2 - 45x + 51$	6	$f(x) = -x^3 + 3x - 7$
7	$f(x) = x^3 - 12x^2 + 45x - 45$	8	$f(x) = -x^3 - 9x^2 - 24x - 18$
9	$f(x) = -x^3 + 15x^2 - 72x + 107$	10	$f(x) = x^3 - 3x^2 + 9$

Завдання 4. Розв'язати систему лінійних рівнянь і перевірити отриманий розв'язок (варіанти в табл. 1.19 і ще один на вибір):

- 1) матричним способом;
- 2) використовуючи функцію **lsolve**;
- 3) використовуючи блок **Given ... Find**;
- 4) використовуючи блок **Given ... Minerr**.

Таблиця 1.19

Варіанти для завдання 4

Варіант	Завдання	Варіант	Завдання
1	$\begin{cases} x - y + 7z = -3, \\ 2x + y - 5z = 0, \\ 3x + 2y - 5z = 1. \end{cases}$	2	$\begin{cases} 10x_1 + x_2 + x_3 = 12 \\ 2x_1 + 10x_2 + x_3 = 13 \\ 2x_1 + 2x_2 + 10x_3 = 14; \end{cases}$
3	$\begin{cases} 2x + 3y - z = 4, \\ x + y - 5z = 1, \\ 3x + y - 3z = -1. \end{cases}$	4	$\begin{cases} x + 2y + z = 3, \\ 3x - y + 2z = -4, \\ 5x + 3y - z = 7. \end{cases}$
5	$\begin{cases} x - 5y + 2z = 9, \\ 3x - y + z = 3, \\ 7x + y - z = -3. \end{cases}$	6	$\begin{cases} 7x - y + 2z = 5, \\ 2x + y - 3z = -7, \\ x - 5y + z = 7. \end{cases}$
7	$\begin{cases} x + y + z = 3, \\ 3x - 2y + z = 2, \\ 5x + 2y - 7z = 0. \end{cases}$	8	$\begin{cases} x - 5y + z = 1, \\ 3x + y - 2z = -7, \\ 2x + 7y + z = 0. \end{cases}$
9	$\begin{cases} 5x - 3y + z = 9, \\ 3x - 7y + 6z = 0, \\ x + 2y + z = 1. \end{cases}$	10	$\begin{cases} x + 2y + 5z = -1, \\ 5x + y - 3z = 5, \\ 7x - 4y - 3z = -5. \end{cases}$

Завдання 5. Знайти розв'язки системи нелінійних рівнянь (варіанти в табл. 1.20).

Таблиця 1.20

Варіанти для завдання 5

Варіант	Завдання	Варіант	Завдання
1	$\begin{cases} y = x^4 - x^2 \\ y = 0,05x - 0,18; \end{cases}$	2	$\begin{cases} y = \sqrt{\frac{x+11}{x}} \\ y = (2x+1)^3; \end{cases}$
3	$\begin{cases} y = 10,5x^2 + 1 \\ y = 5(x+1)^3; \end{cases}$	4	$\begin{cases} y = -3,51x^3 - 6,1x + 2 \\ y = 1,08x - 1,9; \end{cases}$
5	$\begin{cases} y = x^4 + 12 \\ y = (2x+1)^3; \end{cases}$	6	$\begin{cases} y = 2x^3 + 3,8x - 6,2 \\ y = 1,2x - 5; \end{cases}$
7	$\begin{cases} y = x^4 - 15 \\ y = (2x+1)^3 \end{cases}$	8	$\begin{cases} y = 2,8x^3 - 6,2x - 3 \\ y = 2x - 0,12; \end{cases}$
9	$\begin{cases} y = \sin 2x \\ y = \cos(x+1); \end{cases}$	10	$\begin{cases} y = 5(x+1)^3 \\ y = \sqrt{x^2 + 1} - x; \end{cases}$

1.5. Побудова і форматування графіків

Ознайомлення з методами побудови графіків у середовищі пакета MathCAD; набуття практичних навичок роботи в середовищі.

1.5.1. Побудова графіків

Однією з переваг MathCAD є легкість побудови графіків.

У MathCAD можна побудувати графіки двох типів:

- двовимірні графіки (у декартових і полярних координатах):
 - XY графік (**XY Plot**) – декартова система координат;
 - полярний графік (**Polar Plot**) – полярна система координат;
- тривимірні графіки:
 - графік тривимірної поверхні (**Surface Plot**);
 - графік ліній рівня (**Contour Plot**);
 - тривимірна гістограма (**3D Bar Plot**);
 - тривимірна множина точок (**3D Scatter Plot**);
 - векторне поле (**Vector Field Plot**) [1-3].

Існує три способи побудови графіків у системі MathCAD:

1. Через Головне меню **Додати** (Insert), вибравши команду **Графіки** (Graphs) (рис. 1.39) і в списку, що розкривається, вибрати тип графіка або скористатися швидкими клавішами (вони передбачені не для всіх типів графіків).

2. На математичній панелі **Математичні** (Math) викликати панель інструментів **Графік** (Graph) (рис. 1.40) натисканням кнопки із зображенням графіка [1-3].

3. Скористатися швидкими клавішами (вони передбачені не для всіх типів графіків). У табл. 1.21 наведені команди (і швидкі клавіші, якщо є) панелі **Графік**.

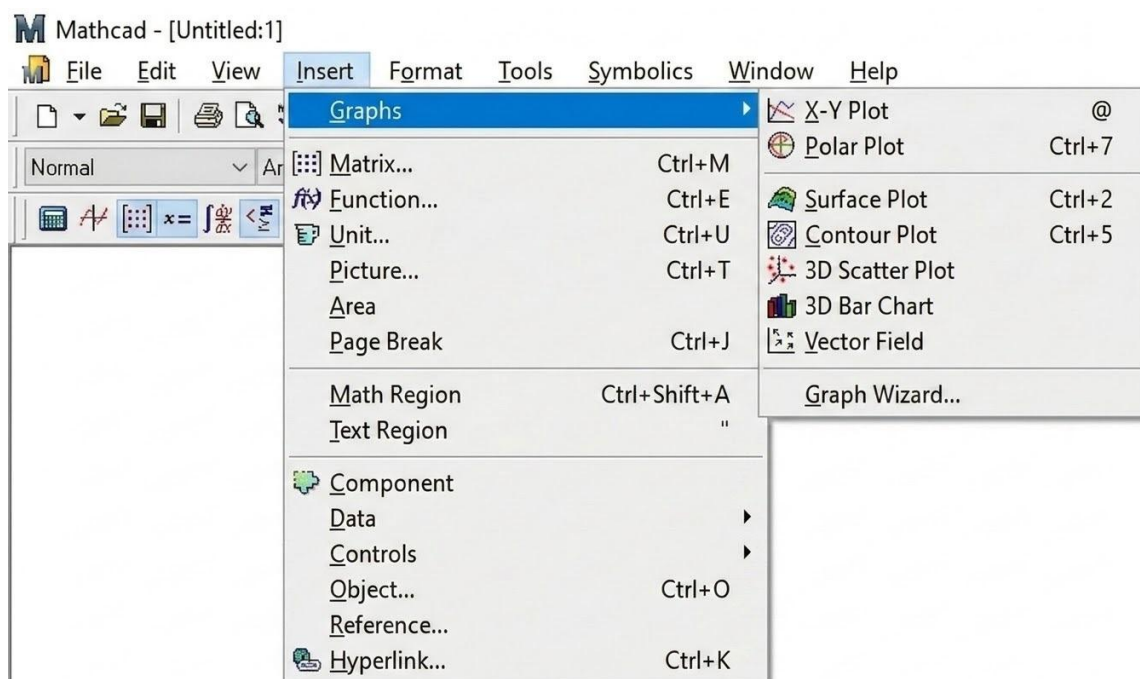


Рис. 1.39. Побудова графіка через головне меню

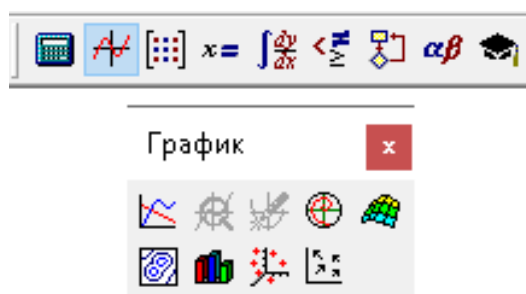


Рис. 1.40. Побудова графіка через панель інструментів **Графік** (Graph)

Таблиця 1.21

Команди панелі **Графік**

Вид	Виконувані дії
1	2
	X-Y Plot (X-Y Графік) клавіша [@]. Служить для побудови графіка функції $y = f(x)$ у вигляді пов'язаних між собою пар координат (x_i, y_i) за заданого інтервалу зміни для i
	Polar Plot (Полярний графік) клавіші [Ctrl + 7]. Служить для побудови графіка функції $r(q)$, заданої в полярних координатах, де полярний радіус r залежить від полярного кута q

1	2
	Surface Plot (Графік поверхні) клавіші [Ctrl + 2]. Служить для подання функції $z = f(x,y)$ у вигляді поверхні в тривимірному просторі. При цьому мають бути задані вектори значень x_i і y_j , а також визначена матриця вигляду $A_{i,j} = f(x,y)$. Ім'я матриці A вказують, заповнюючи рамку-шаблон. За допомогою цієї команди можна будувати параметричні графіки
	Contour Plot (Графік контуру). Буде діаграму ліній рівня функції вигляду $z = f(x,y)$, тобто відображає точки, у яких ця функція набуває фіксованого значення $z = \text{const}$
	3D Scatter Plot (Графік 3D точковий). Служить для точкового подання матриці значень $A_{i,j}$ або відображення значень функції $z = f(x,y)$ у заданих точках. Цю команду можна також використовувати для побудови просторових кривих
	3D Bar Plot (Графік 3D діаграми). Служить для подання матриці значень $A_{i,j}$ або відображення значень функції $z = f(x,y)$ у вигляді тривимірної стовпчикової діаграми
	Vector Field Plot (Графік векторного поля). Служить для відображення двовимірних векторних полів $V = (V_x, V_y)$. При цьому компоненти векторного поля V_x і V_y мають бути подані у вигляді матриць. За допомогою цієї команди можна побудувати поле градієнта функції $f(x,y)$
	3D Plot Wizard (виклик майстра для швидкої побудови тривимірного графіка через меню Додати). Вибираючи цю команду, виникає ряд спливаючих вікон, у яких потрібно вибрати параметри побудови тривимірного графіка (задають тип тривимірного графіка, стиль його зображення, колірна гама). Графік за замовчуванням будують на проміжку від -5 до +5

Взагалі виклик будь-якої з цих команд, крім останньої, призведе до вставлення в робочий документ спеціальної графічної області, яку називають *шаблоном*.



Шаблон може містити одне або кілька полів введення. Заповнення цих полів відповідними вихідними даними завершить процес побудови графіка.

1.5.2. Основні правила побудови графіків:

- натиснути мишею там, де потрібно створити графік;
- будь-яким способом вибрати тип графіка або натиснути відповідну комбінацію клавіш, з'явиться шаблон з шістьма порожніми полями, які потрібно заповнити;
- порожнє поле в середині горизонтальної осі призначене для незалежної змінної. Туди вводять дискретну змінну, змінну з індексом тощо;
- порожнє поле в середині вертикальної осі призначене для змінної, графік якої потрібно побудувати. Туди вводять дискретну змінну, змінну з індексом або функцію, що містить змінну, яка є **на горизонтальній осі**;
- інші чотири поля призначені для зазначення діапазонів, їх заповнюють за замовчуванням або вручну;
- для відображення графіка потрібно натиснути мишею поза полем або **F9**;
- щоб показати кілька залежностей на одному графіку, ввести першу змінну по осі ординат із комою наприкінці. Нижче з'явиться порожнє поле для другої змінної (виразу), ввести другу змінну з комою наприкінці, нижче з'явиться третє поле і т. д. (до 16 графіків) [1-3].

1.5.3. Побудова двовимірних графіків

Двовимірні графіки в декартовій системі координат

Побудову розглянемо на прикладі двовимірного графіка $f(x) = x^2$ (рис. 1.41).

Введемо значення аргументу x як ранжовану змінну. Запишемо функцію залежно від цього аргументу. Зазначимо місце, де буде розташована область графіка. Виберемо на панелі **График** позначку  **X-Y График**. У позначеному місці документа з'явиться область графіка з

кількома місцями-заповнювачами. У місця заповнень на осях ввести імена змінних і функцій, які мають бути зображені на графіку [1-3].

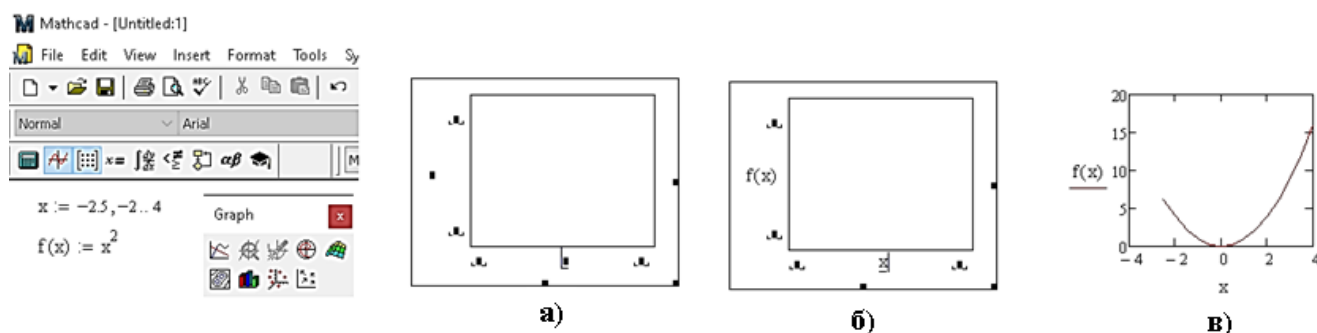


Рис. 1.41. Приклад побудови графіка: *а* – визначення функції, її аргументів і виклик шаблона для побудови графіка; *б* – заповнення полів даними; *в* – ілюстрація результату

Вид графіка можна змінити, змінюючи його дані, форматуючи його зовнішній вигляд або додаючи елементи оформлення. Для форматування графіків можна скористатися контекстним меню, вибравши **Format** (або натиснути на тій області, форму якої заплановано змінити).

На одному рисунку (графіку) можна побудувати до 16 графіків (залежностей). Для цього *через кому* в місце заповнення для найменування осі ординат (вісь y) вносять найменування функцій (рис. 1.42).

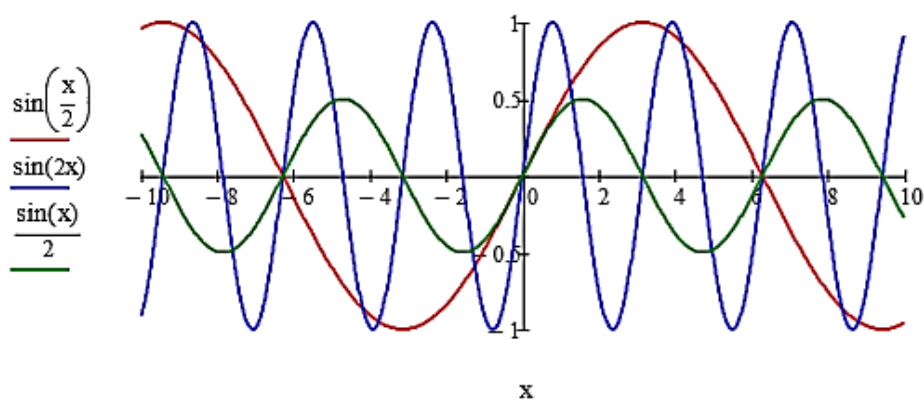


Рис. 1.42. Побудова декількох графіків на одному рисунку

1.5.3.2. Криві на площині, задані параметрично. Рівняння $x = x(t)$, $y = y(t)$, де $x(t)$ і $y(t)$ безперервні, якщо t із (a, b) , що встановлюють залежність декартових координат (x, y) точки площини від значення параметра t , визначають на площині криву, задану в параметричній формі (рис. 1.43, 1.44).

Приклад 1. Побудувати графік функції, яка задана параметрично

$$\begin{cases} x = \cos^3 t \\ y = \sin^3 t \end{cases}$$

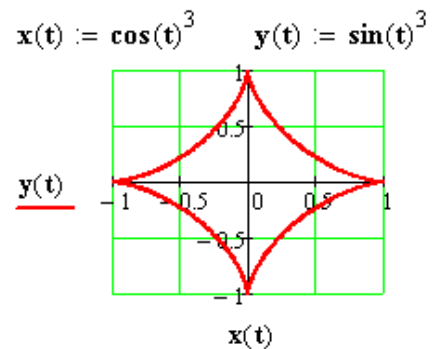


Рис. 1.43. Побудова графіка функції, яка задана параметрично

Приклад 2. Для побудови графіків функцій, заданих параметрично у вигляді вектора, замість незалежної змінної x під віссю абсцис необхідно задати індексовану змінну x_i . А поруч з віссю ординат необхідно відповідно вказати y_i .

Для нанесення на графік функції окремих точок їхні координати вказують через кому під віссю абсцис і зліва від осі ординат. Для виведення множини точок формують два вектори, один із яких містить абсциси точок, а інший – їхні ординати. Тоді на графіку у відповідних заповнювачах вказують тільки імена векторів.

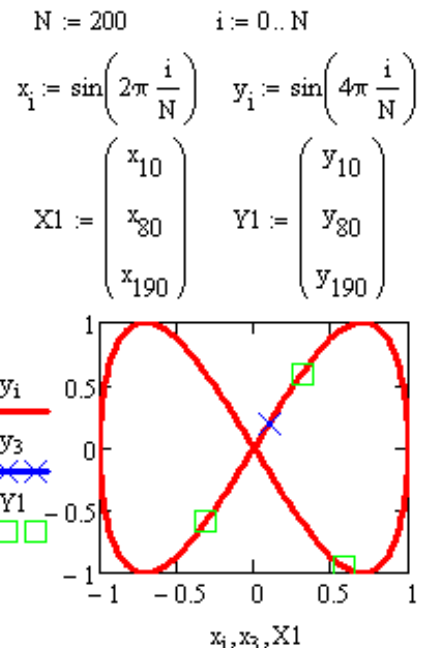


Рис. 1.44. Побудова графіка функції, яка задана параметрично у вигляді вектора

1.5.3.3. Побудова графіків у полярній системі координат. Вставити

шаблон для побудови полярного графіка можна, якщо натиснути кнопку  на панелі графіків **Графік** (Graph) або іншим способом. Полярні графіки будують аналогічно декартовим. Різниця полягає лише в тому, що в нижній частині шаблону вводять значення полярного кута, а ліворуч – полярного радіуса (рис. 1.45).

Приклад 3. Побудова графіків $\rho(\varphi) := 2 - \cos(2\varphi)$ у полярних координатах.

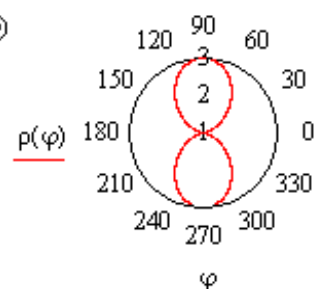


Рис. 1.45. Побудова графіка в полярній системі координат

1.5.3.4. Поверхні, отримані обертанням кривих навколо осей.

Цікаві об'ємні фігури можна отримати, обертаючи деяку криву навколо тієї чи іншої осі. Побудова цих фігур обертання схожа на побудову параметрично заданих поверхонь (рис. 1.46, 1.47).

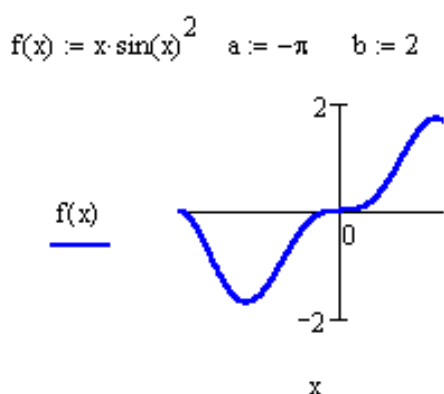


Рис. 1.46. Поверхні, отримані обертанням кривих навколо осей

При цьому необхідно забезпечити перерахунок координат точок фігури за відомими з геометрії формулами. У MathCAD вбудована функція **CreateMesh**, за допомогою якої можна побудувати параметрично задані поверхні. Забігаючи наперед, тут для об'ємних фігур потрібно скористатися кнопкою  **Surface Plot** на панелі графіків (рис. 1.40).

Обертання кривої навколо осі Ox

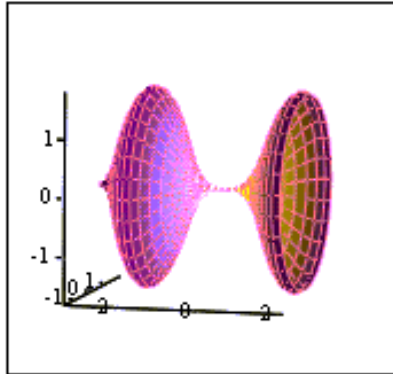
$$\begin{aligned} F(u, v) &:= u & \text{mesh} &:= 30 \\ G(u, v) &:= f(u) \cdot \cos(v) \\ H(u, v) &:= f(u) \cdot \sin(v) \end{aligned}$$

$$SX := \text{CreateMesh}(F, G, H, a, b, 0, 2\pi, \text{mesh})$$

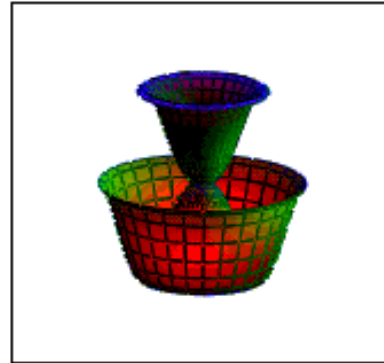
Обертання кривої навколо осі Oy

$$\begin{aligned} X(u, v) &:= u \cdot \sin(v) \\ Y(u, v) &:= u \cdot \cos(v) \\ Z(u, v) &:= f(u) \end{aligned}$$

$$SY := \text{CreateMesh}(X, Y, Z, a, b, -\pi, \pi, \text{mesh})$$



SX



SY

Рис. 1.47. Поверхні, отримані обертанням кривих навколо осей

1.5.4. Побудова тривимірних графіків

1.5.4.1. Побудова графіка функції $z = f(x, y)$ у вигляді поверхні.

Для побудови графіка поверхні (рис. 1.48) можна скористатися двома способами:

- якщо потрібно тільки подивитися загальний вигляд поверхні, то MathCAD дає змогу швидко побудувати подібні графіки. Для цього достатньо визначити функцію $f(x, y)$ або матрицю, де задано розподілення даних на площині X, Y , і виконати команду **Додати (Insert) → Графік (Graph) → Графік поверхні (Surface Plot)**;

- натиснути відповідну кнопку **Графік** (клавіші **[Ctrl + 2]**).

У графічній області, що з'явилася, під осями на місці шаблону для введення вказати ім'я (без аргументів) функції або ім'я матричної змінної. MathCAD автоматично побудує графік поверхні. Незалежні змінні x і y набувають значення з інтервалу $[-5, 5]$ $[1-3]$.

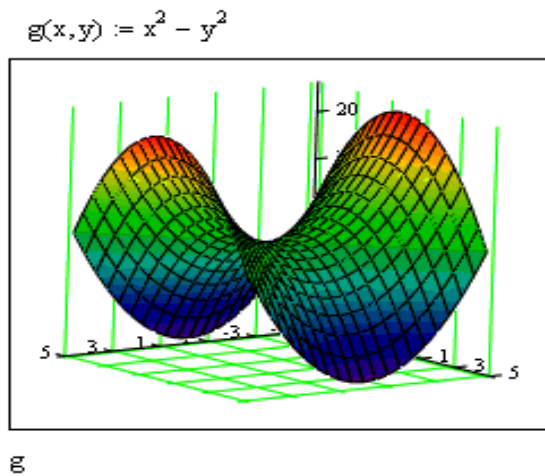


Рис. 1.48. Графік функції у вигляді поверхні

За необхідності цей інтервал може бути зменшений або збільшений. Для цього необхідно виділити графік і скористатися командою **Формат** (Format) → **Графік** (Graph) → **3D-Графік** (3D Plot) або натиснути правою кнопкою миші по виділеному графіку і в контекстному меню вибрати команду **Формат** (Format). У вікні **Формат 3D-графіка** (3D Plot Format) на вкладці **Дані швидкого графіка** (QuickPlot Data) можна встановити інші параметри незалежних змінних x і y . Відповідно зміниться і вигляд графіка.

1.5.4.2. Побудова графіка поверхні у визначеній області зміни незалежних змінних або з конкретним кроком їх зміни. Необхідно спочатку встановити вузлові точки x_i і y_j , у яких будуть визначені значення функції. Після (а можна і до) цього визначити функцію $f(x, y)$, графік якої потрібно побудувати. Після цього необхідно сформувати матрицю значень функції у вигляді $A_{i,j} = f(x_i, y_j)$ (рис. 1.49).

Тепер після виконання команди **Вставка** (Insert) → **Графік** (Graph) → **Графік площини** (Surface Plot) в графічній області, що з'явилася, досить ввести ім'я матриці (без індексів).

```

N := 15
i := 0..N      j := 0..N
x_min := -1.5  x_max := 1.5
y_min := -1.5  y_max := 1.5

x_i := x_min +  $\frac{i}{N} \cdot (x_{max} - x_{min})$ 
y_j := y_min +  $\frac{j}{N} \cdot (y_{max} - y_{min})$ 
f(x, y) := sin(x) · exp(-x2 - y2)
Ai,j := f(xi, yj)

```

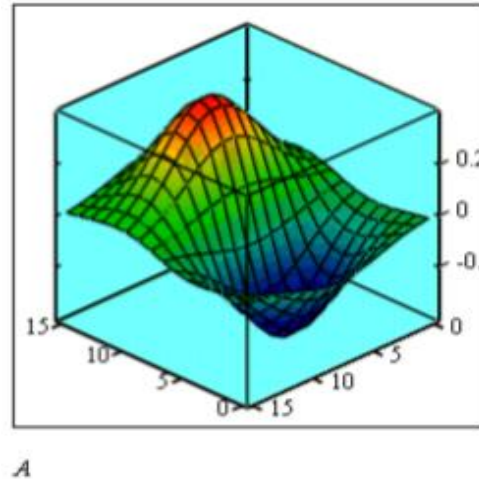


Рис. 1.49. Графік поверхні у визначеній області

1.5.4.3. Побудова графіка ліній рівня. Для побудови графіка ліній рівня функції необхідно зробити так само, як це було описано вище (рис. 1.49), тільки замість команди (Графік Площини) вибрати команду **Графік контуру** (Contour Plot) (рис. 1.50).

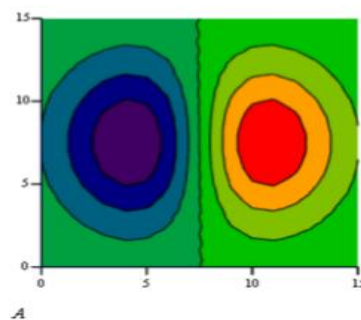
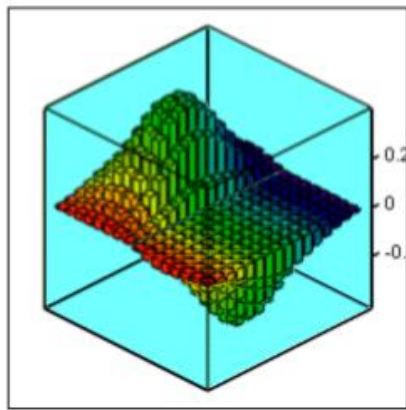


Рис. 1.50. Графік ліній рівня

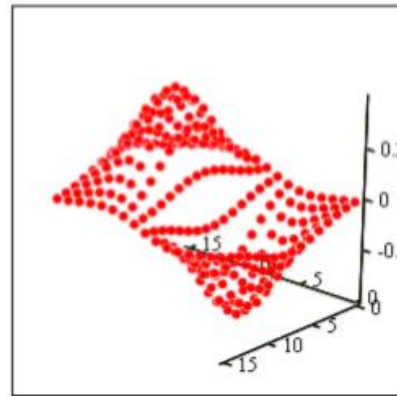
1.5.4.4. Побудова тривимірного стовпчикового і точкового графіків (рис. 1.51). За допомогою команди **3D Bar Plot** (3D Діаграми) можна побудувати тривимірний стовпчиковий графік даної функції (рис. 1.51, а).

За допомогою команди **3D Scatter Plot** (3D Точечний) - тривимірний точковий графік (рис. 1.51, б).



A

а



A

б

Рис. 1.51. Стовпчиковий і точковий графіки

1.5.4.5. Побудова графіка поверхні, заданої параметрично. Якщо поверхня задана параметрично, це означає, що всі три координати – x , y і z – задані як функції від двох параметрів u і v . Спочатку необхідно задати вектори значень параметрів u_i і v_j . Потім визначити матриці значень функцій координат $x(u,v)$, $y(u,v)$ і $z(u,v)$ (рис. 1.52).

Після вибору команди  **Surface Plot** у документі з'явиться графічна область. У вільній клітинці внизу області треба вказати в дужках імена (без аргументів та індексів) трьох матриць – x , y , z .

$$N := 30 \quad i := 0..N \quad j := 0..N$$

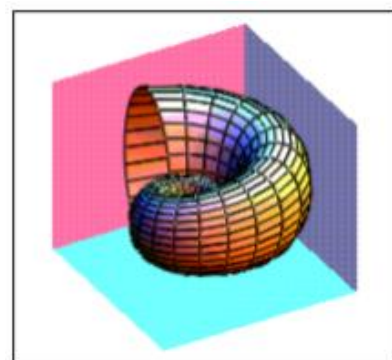
$$\phi_i := 3 \cdot \pi \cdot \frac{i}{N} \quad \psi_j := 2 \cdot \pi \cdot \frac{j}{N}$$

$$R(\phi) := 8 \cdot e^{-\frac{\phi}{5}} \quad r(\phi) := 4 \cdot e^{-\frac{\phi}{5}}$$

$$x_{i,j} := (R(\phi_i) + r(\phi_i) \cdot \cos(\psi_j)) \cdot \cos(\phi_i)$$

$$y_{i,j} := (R(\phi_i) + r(\phi_i) \cdot \cos(\psi_j)) \cdot \sin(\phi_i)$$

$$z_{i,j} := r(\phi_i) \cdot \sin(\psi_j)$$



(x, y, z)

Рис. 1.52. Графік поверхні параметричної функції, яка залежить від двох параметрів

1.5.5. Форматування графіків

1.5.5.1. Редагування графіків у декартовій системі координат.

Якщо не влаштовує зовнішній вигляд побудованих графіків, можна його змінити, виділивши графік (натиснувши так, щоб навколо нього з'явилася рамка) і скориставшись командою **Формат (Format) → Графік (Graph) → Графік ХУ (XY Plot)** або виконавши на графіку натискання правою кнопкою миші і вибравши команду **Формат (Format)** із контекстного меню (можна виконати також подвійне натискання лівою кнопкою миші). У результаті на екрані з'явиться діалогове вікно **Форматування вибраного графіка (Formatting Currently Selected X-Y Plot)**, що дає змогу змінити вигляд графіка (рис. 1.53) [1-3].

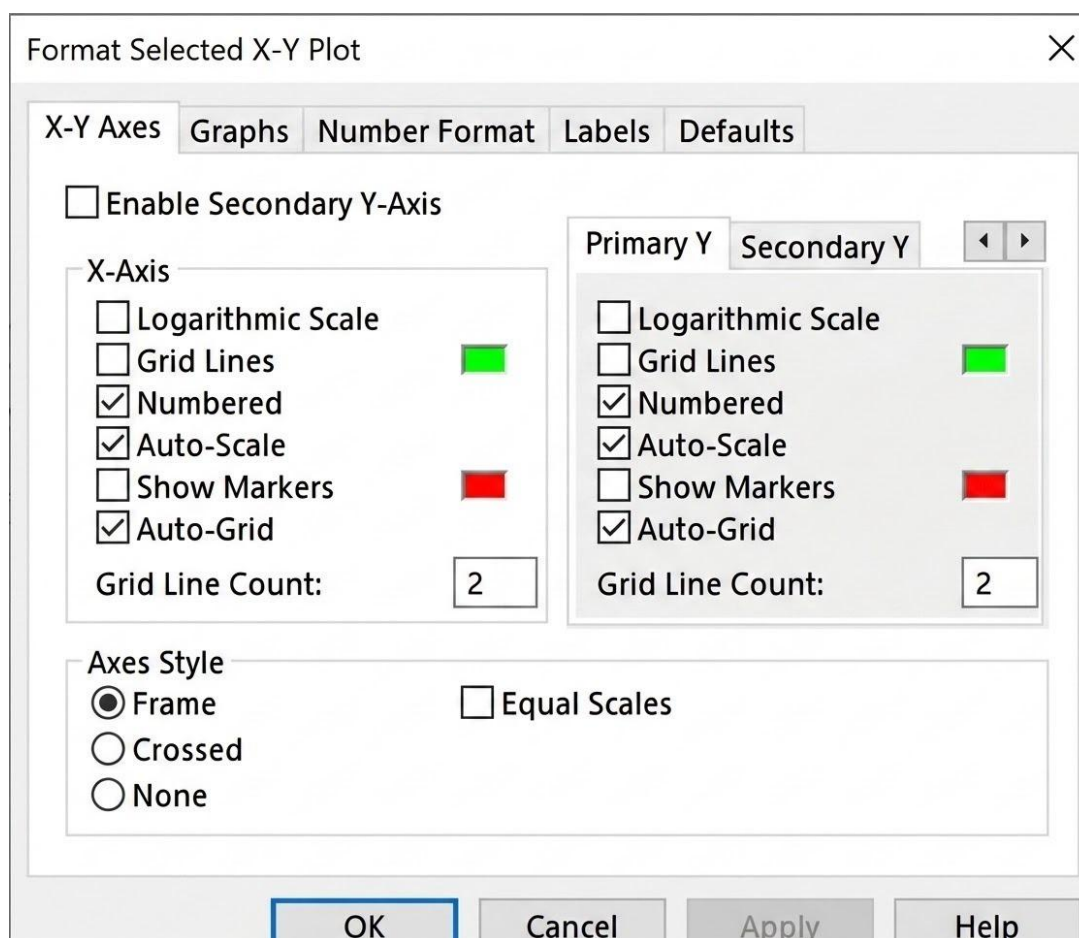


Рис. 1.53. Діалогове вікно **Форматування вибраного графіка**

Це діалогове вікно містить кілька вкладок: **Осі X, Y** (X-Y Axes); **Трасування** (Traces); **Формат числа** (Number Format), **Підписи** (Labels), **За замовчуванням** (Defaults). Призначення вкладок наведене в табл. 1.22 [1-3].

Таблиця 1.22

Вкладки діалогового вікна **Форматування вибраного графіка**

Вкладка	Дії, які можна виконувати
1	2
Осі X, Y (X-Y Axes)	<i>Дає змогу формувати осі координат:</i> • логарифмічний масштаб (Log Scale) – задає логарифмічні осі, тут межі графіка задані додатними числами; • лінії сітки (Grid Lines) – задає відображення сітки з паралельних осей ліній; • нумерація (Numbered) – задає відображення підпису до маркування на осях; • автомасштабування (Autoscale) – задає автоматичне знаходження відповідних меж для осей. Але якщо самому задати у відповідних заповнювачах мінімальні і максимальні значення x_{min}, x_{max}, y_{min}, y_{max}, саме ці значення будуть використані для визначення меж графіка; • показати мітки (Show Markers) – якщо встановити цю опцію, то в графічній області з'являться чотири додаткові комірки для створення червоних ліній маркування, що відповідають двом спеціальним значенням x і двом спеціальним значенням y; • автосітка (Auto Grid) – для установлення цієї опції кількість ліній сітки визначає MathCAD
Трасування (Traces)	<i>Для зміни типу ліній графіків:</i> • значення в легенді (Legend Lable) – кожній кривій задано ім'я, так звана легенда. Легенда відображена в нижній частині графічної області, а поруч із кожною легендою відображений тип лінії відповідної кривої; • символ (Symbol) – дає змогу вибрати символ для кожної точки кривої (плюс, хрестик, кружок та ін.); • лінія (Line) – можна вибрати тип лінії; • колір (Color) – задавання кольору подання кривої на екрані;

1	2
	• тип (Type) – дає змогу вибрати один із семи видів графіка: у вигляді кривих (), стовпців () і т. д.; • опції (Line Weight) – дає змогу задавати товщину ліній графіка. <i>У нижній частині вкладки Трасування (Traces) розташовані опції:</i> • приховати аргументи (Hide Arguments) – ця опція за замовчуванням відключена. У цьому випадку під ім'ям функції поряд із віссю ординат вказано поточний тип ліній. Якщо встановити цю опцію, вказівка типу ліній зникне; приховати легенду (Hide Legend) – за замовчуванням легенда не відображена. Якщо потрібно відобразити під графіком текст легенди, його необхідно перед цим ввести в поле Значення в легенді (Legend Label) і підтвердити введення, натиснувши на кнопку Застосувати
Формат числа (Number Format)	<i>Дає змогу задавати формат числа</i>
Підписи (Labels)	<i>Дає змогу ввести заголовок графіка і підписи для осей</i>
За замовчуванням (Defaults)	<i>Дає змогу задавати формат числа</i>

1.5.5.2. Форматування тривимірних графіків. За необхідності зовнішній вигляд створеного тривимірного графіка можна змінити, виконавши команду **Формат** (Format) → **Графік** (Graph) → **3D-Графік** (3D Plot) або двічі натиснути мишкою на графічній області. У результаті на екрані з'явиться діалогове вікно **Формат 3D-графіка** (3-D Plot Format), що дає змогу змінювати параметри відображення графіка. Розглянемо тут основні опції. З усіма тонкощами управління видом графіка розібратися можна самостійно, побудувавши графік і проєкспериментувавши, вибираючи ті чи інші опції.

Діалогове вікно **Формат 3D-графіка** (3D Plot Format) містить кілька вкладок (рис. 1.54). Деякі з них розглянемо більш детально, а для інших – опишемо лише їхнє функціональне призначення.

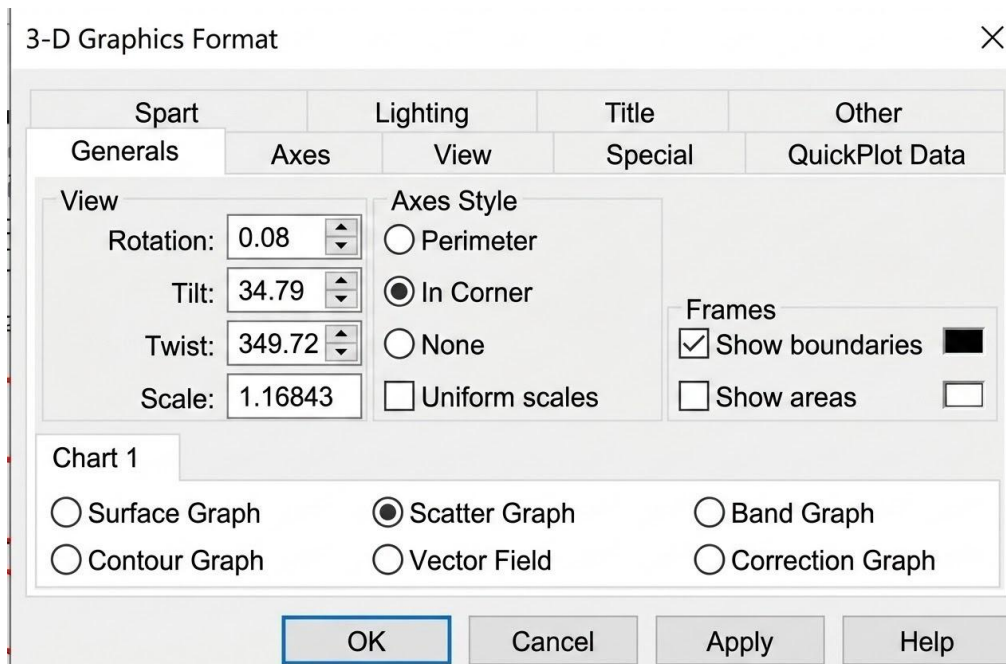


Рис. 1.54. Діалогове вікно **Форматування вибраного графіка**

Короткий опис опцій діалогового вікна **Формат 3D-графіка (3-D Plot Format)** наведений у табл. 1.23.

Таблиця 1.23

Вкладки діалогового вікна **Формат 3D-графіка (3-D Plot Format)**

Вкладка	Дії, які можна виконувати
1	2
Загальне (General)	<i>Дає змогу задати загальний вигляд системи координат:</i> - в області Огляд (View) задати напрямок погляду спостерігача на тривимірний графік. Значення в полі Поворот (Rotation) визначає кут повороту навколо осі Z в площині X-Y. Значення в полі Нахил (Tilt) задає кут нахилу лінії погляду до площини X-Y. Поле Масштаб (Zoom) дає змогу збільшити (зменшити) графічне зображення в зазначену кількість разів; - в області Стиль осей (Axes Style) задати вид осей, вибравши селекторну кнопку Периметр (Perimetr) – осі завжди знаходяться на передньому плані, або Кут (Corner) – точка перетину осей Ox і Oy задана елементом $A_{0,0}$ матриці A;

1	2
	• в області Рамки (Frames) опція Показати поле (Show box) призначена для відображення навколо графіка куба з прозорими гранями, а опція Показати границю (Show border) дає змогу взяти графік у прямокутну рамку; • в області Графік 1 (Plot 1) є селекторні кнопки для подання графіка в інших виглядах (контурний, точковий, векторне поле та ін.)
Осі (Axes)	<i>Дає змогу змінювати зовнішній вигляд осей координат:</i> • опції області Сітка (Grids) дають змогу відобразити на графіку лінії, описані рівняннями $x, y, z = \text{const}$; • опції Показати числа (Show Numbers) відображають мітки на осях і підписи до них. При цьому поряд з осями Ox і Oy вказують не значення вузлових точок x_i, y_j, а значення індексів i і j, у той час як вісь Oz розмічена відповідно до інтервалу, якому належать елементи матриці значень A_{ij}; • опція Автосітка (Auto Grid) – установлення відстані між сусідніми відмітками на осях самостійно програмою, чи користувач сам вказує кількість ліній сітки, якщо ця опція відключена; • опція Автомасштабування (Auto Scale) – установлення межі побудови графіка і масштаби по осях автоматично, чи користувач самостійно задає межі зміни змінних у полях Мін. значення (Minimum Value) і Макс. значення (Maximum Value), якщо ця опція відключена
Оформлення (Appearance)	Дає змогу змінювати для кожного графіка вид і колір заливки поверхні (область Fill Options); вид, колір і товщину додаткових ліній на графіку (область Line Options); наносити на графік точки даних (опція Draw Points області Point Options), змінювати їхній вид, розмір і колір
Підсвічування (Lighting)	З увімкненням опції Увімкнути підсвічування (Enable Lighting) можна вибрати колірну схему для освітлення, «встановити» кілька джерел світла, вибравши для них колір освітлення і визначивши його напрямок
Задні плани (Backplanes)	Дає змогу змінити зовнішній вигляд площин, що обмежують область побудови: колір, нанесення сітки, визначення її кольору і товщини, промальовування кордонів площин
Спеціальна (Special)	Дає змогу змінювати параметри побудови, специфічні для різних типів графіків

1	2
Додатково (Advanced)	Дає змогу встановити параметри друку і змінити колірну схему для встановлення кольору поверхні графіка, а також вказати напрямок зміни забарвлення (уздовж осі Ox , Oy або Oz). Включення опції Наявність Туману (Enable Fog) робить графік нечітким, злегка розмитим (напівпрозорим). З увімкненням опції Перспектива (Perspective) з'являється можливість вказати у відповідному полі відстань до спостерігача

1.5.6. Крива в просторі

Тривимірні точкові графіки можна використовувати для побудови зображення просторових кривих. Просторові криві задані зазвичай у вигляді $x(t)$, $y(t)$, $z(t)$, де t являє собою безперервний дійсний параметр (рис. 1.55).



Рис. 1.55. Побудова гвинтової кривої

Оскільки з побудовою тривимірної точкової діаграми MathCAD дає змогу відображати на графіку тільки окремі точки, то, з'єднуючі їхні лінії, необхідно спочатку визначити три вектори координат: x_i , y_i , z_i .

Просторова крива створювана командою **Додати** (Insert) → **Графік** (Graph) → **3D-графік розкиду** (Scatter Plot). Можна використовувати панель **Графік** (Graph), вибравши відповідну піктограму . Для з'єднання точок необхідно на вкладці **Оформлення**

(Appearance) вікна форматування графіків вказати опцію **Лінії** (Line).

1.5.7. Векторні і градієнтні поля

Команда **Додати** (Insert) → **Графік** (Graph) → **Векторне поле** Vector Field Plot) служить для подання двовимірних векторних полів $\mathbf{v} = (v_x, v_y)$ (рис. 1.56).

При цьому векторне поле необхідно спочатку визначити як вектор-функцію двох координат – x і y . Потім задають вектори значень вузлових точок x і y . За допомогою цих векторів компоненти векторного поля $v_x(x,y)$ і $v_y(x,y)$ будуть генеровані у вигляді матриць значень $v_{x_{i,j}}$ і $v_{y_{i,j}}$.

$$v(x,y) := \begin{pmatrix} \frac{-y}{\sqrt{x^2 + y^2}} \\ \frac{x}{\sqrt{x^2 + y^2}} \end{pmatrix} \quad \begin{array}{l} N := 20 \\ i := 0..N-1 \\ j := 0..N-1 \end{array}$$

$$a := -2 \quad b := 2 \quad x_i := a + \frac{i}{N-1} \cdot (b - a)$$

$$c := -2 \quad d := 2 \quad y_j := c + \frac{j}{N-1} \cdot (d - c)$$

$$V_{i,j} := v(x_i, y_j)$$

$$v_{x_{i,j}} := (V_{i,j})_0 \quad v_{y_{i,j}} := (V_{i,j})_1$$

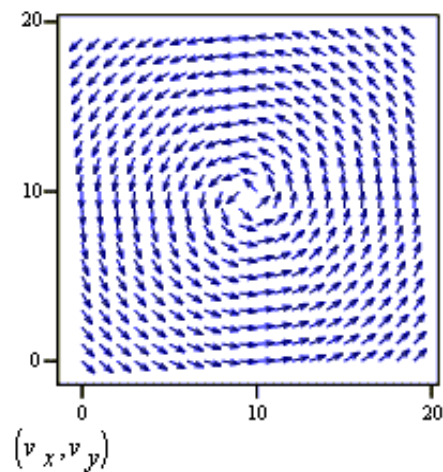


Рис. 1.56. Побудова векторного поля

Так само можна побудувати градієнтне поле скалярної функції $f(x,y)$. Градієнтне поле для функції двох змінних є двовимірним векторним полем (рис. 1.57).

Як і в інших випадках, зовнішній вигляд зображення векторного поля можна легко змінити, двічі натиснувши в області графіка і змінивши необхідні параметри в діалоговому вікні.

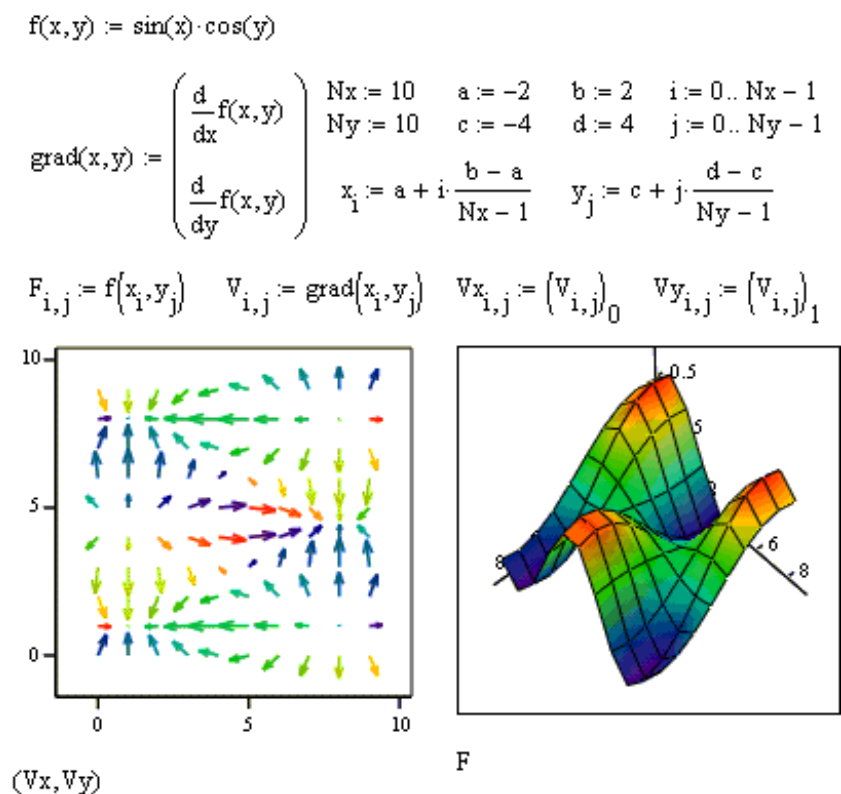


Рис. 1.57. Побудова градієнтного поля

ЗАВДАННЯ ДЛЯ ВИКОНАННЯ

Завдання 1. Розв'язати графічно квадратне рівняння $ax^2 + bx + c = 0$ для значень a, b, c із табл. 1.24. Визначити корені.

Таблиця 1.24

Варіанти завдання 1

Варіант	a	b	c
1	2	-8	-3
2	-1,1	-7	1
3	1	5	-1,18
4	-1,3	-6	1,2
5	-7	15	2
6	-5	-10	3,3
7	3	11	-3,1
8	-2,3	-9	2,5
9	2,5	8	-1,9
10	3,8	-12	-4,5

Перевірити визначені корені рівняння, розв'язавши його за допомогою функцій **root** і **polyroots**, зробивши відповідну підстановку у вихідне рівняння.

Завдання 2. Побудувати ХУ-графік функції. Варіант вибрати із табл. 1.25. Відформатувати отриманий графік: змінити товщину ліній, колір тощо. На цьому ж графіку побудувати ще один на вибір.

Завдання 3. Побудувати параметричний графік системи рівнянь. Варіант вибрати з табл. 1.25. Відформатувати отриманий графік: змінити товщину ліній, колір тощо. На цьому ж графіку побудувати ще один на вибір.

Завдання 4. Побудувати полярний графік функції. Варіант вибрати з табл. 1.25. Відформатувати отриманий графік: змінити товщину ліній, колір тощо. На цьому ж графіку побудувати ще один на вибір [1-3].

Таблиця 1.25

Варіанти завдань

Варіант	ХУ-графік	Параметричний графік	Полярний графік
1	2	3	4
1	$f(x) = \frac{x^3}{x^2 - 9}$	$\begin{cases} x = 2t + 1 \\ y = 4t^2 - 1 \end{cases}$	$\rho = 3 \sin 3 \left(\varphi + \frac{\pi}{6} \right)$
2	$f(x) = x^2 + \frac{2}{x}$	$\begin{cases} x = t^3 \\ y = t^2 - 2 \end{cases}$	$\rho = 4 \cos 2 \left(\varphi + \frac{\pi}{4} \right)$
3	$f(x) = \frac{x}{x^2 - 4}$	$\begin{cases} x = 2t - 1 \\ y = 1 - 4t^2 \end{cases}$	$\rho = 3 \sin 3 \left(\varphi + \frac{\pi}{3} \right)$
4	$f(x) = \frac{x^3 + 4}{x^2}$	$\begin{cases} x = t^2 \\ y = \frac{t^3}{2} \end{cases}$	$\rho = 2 + \sin 2 \left(\varphi + \frac{\pi}{4} \right)$
5	$f(x) = \frac{4x - 12}{(x - 2)^2}$	$\begin{cases} x = 2 \cos^2 t \\ y = 3 \sin^2 t \end{cases}$	$\rho = 3 - 2 \cos 3\varphi$
6	$f(x) = \frac{x}{\sqrt[3]{x^2 - 1}}$	$\begin{cases} x = t^2 \\ y = \frac{t^3}{3} \end{cases}$	$\rho = 3 + 2 \cos \frac{\varphi}{3}$

1	2	3	4
7	$f(x) = \frac{x}{\sqrt[3]{(x-2)^2}}$	$\begin{cases} x = 2 \cos t \\ y = 3 \sin^2 t \end{cases}$	$\rho = 2 - 6 \cos 3\varphi$
8	$f(x) = \frac{x^2 + 1}{x^2 - 1}$	$\begin{cases} x = \operatorname{tg} t \\ y = \cos^2 t \end{cases}$	$\rho = 2 + 4 \cos \frac{\varphi}{2}$
9	$f(x) = \sqrt{\frac{x^3}{x-2}}$	$\begin{cases} x = 2 \cos^4 t \\ y = 2 \sin^4 t \end{cases}$	$\rho = 3 - 2 \sin \frac{\varphi}{4}$
10	$f(x) = \frac{x^3}{(x-2)^2}$	$\begin{cases} x = \frac{e^t + e^{-t}}{2} \\ y = \frac{e^t - e^{-t}}{2} \end{cases}$	$\rho = 3 + 2 \sin 3 \left(\varphi + \frac{\pi}{2} \right)$

Завдання 5. Побудувати тривимірні графіки різних типів, використовуючи для цього функцію залежності $g(x, y) = 0,1x^3 + y^2$ і

матрицю $z := \begin{pmatrix} 3 & 8 & 0.5 & 4.1 & 1.22 \\ 9 & 12 & 0 & 2.1 & 1.5 \\ 1.5 & 3.3 & 2.5 & 9 & 2 \\ 13 & 11 & 15 & 4.1 & 2.9 \\ 4 & 3 & 6.5 & 4 & 4 \end{pmatrix}$ і відформатувати їх (підписати, зробити

кольоровими тощо):

- графік поверхні, заданий функцією;
- графік поверхні, заданий матрицею;
- швидко побудувати графік ліній рівня функції;
- графік ліній рівня, заданий матрицею;
- графік тривимірної гістограми;
- швидко побудувати графік тривимірної гістограми функції;
- графік тривимірної гістограми, заданий матрицею;
- швидко побудувати графік множини точок функції;
- графік множини точок, заданий матрицею.

1.6. Анімація в MATHCAD

Набуття навичок створення, відтворення і побудови роликів анімації в MathCAD-документі.

У багатьох випадках **найбільш видовищний спосіб подання результатів математичних розрахунків – це анімація**. MathCAD дає змогу створювати анімаційні ролики і зберігати їх у відеофайли.

Основний принцип анімації в MathCAD – **покадрова анімація**.

Ролик анімації – це просто послідовність кадрів, що являють собою деяку ділянку документа, який виділяє користувач [1-3].

Розрахунки проводять окремо для кожного кадру, причому формули і графіки, які в ньому містяться, мають бути функцією від номера кадру. Номер кадру задано системною змінною **FRAME**, яка може *набувати лише натуральних значень*. За замовчуванням, якщо не ввімкнений режим підготовки анімації, **FRAME = 0**.

Єдине призначення системної змінної **FRAME** – це управління анімацією.

1.6.1. Створення анімаційного кліпу

Розглянемо послідовність дій для створення ролика анімації, який, наприклад, демонструє переміщення гармонійної хвилі, що біжить. При цьому кожен момент часу буде заданий змінною FRAME.

1. Ввести в документ необхідні вирази і графіки, у яких бере участь змінна номера кадру FRAME.

Підготувати частину документа, для якої створюють ролик анімації, так, щоб вона перебувала в полі зору на екрані [1-3].

Для нашого прикладу підготовка зводиться до визначення функції $f(x,t):=\sin(x-t)$ і створення її Декартового графіка $y = (x, FRAME)$ (рис. 1.57).

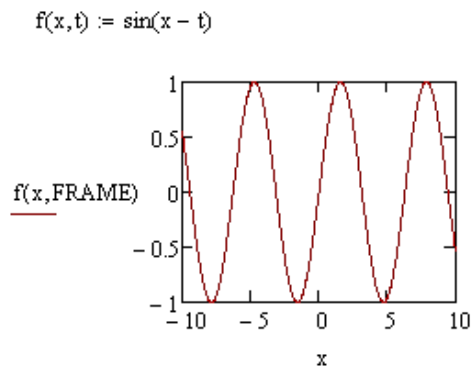


Рис. 1.57. Підготовка до створення ролика анімації

2. Виконати команду **Інструменти > Анімація > Запис** (Tools > Animation > Record) (рис. 1.58).

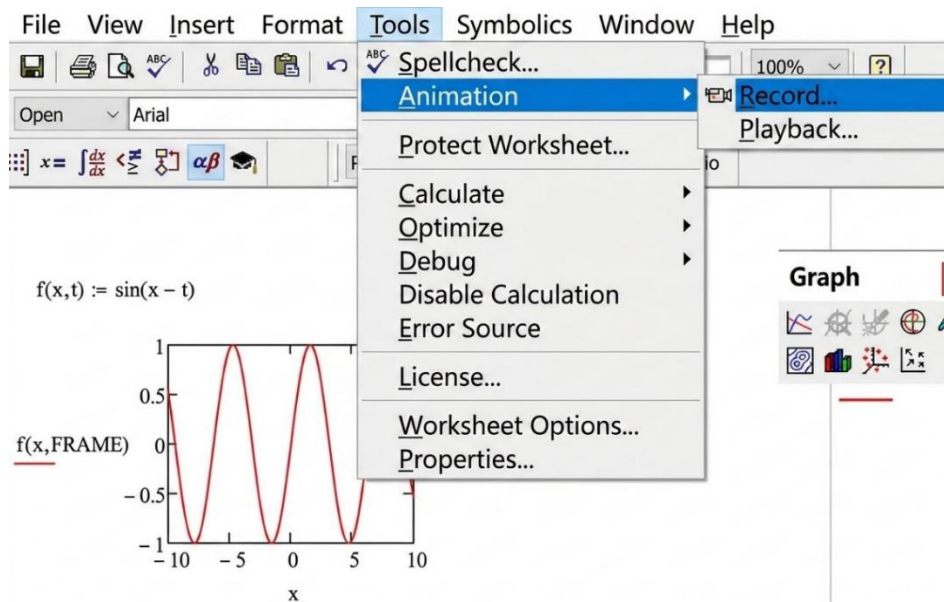


Рис. 1.58. Виклик запису ролика анімації

3. У діалоговому вікні **Запис анімації** (Record Animation) задати номер першого кадру в поле **С** (From), номер останнього кадру в поле **По** (To) і швидкість анімації в поле **На** (At) у кадрах за секунду [1-3].

4. Виділити протяганням покажчика миші з натиснутою лівою кнопкою миші побудований графік, що буде роликом, у маркувальний прямокутник (рис. 1.59).

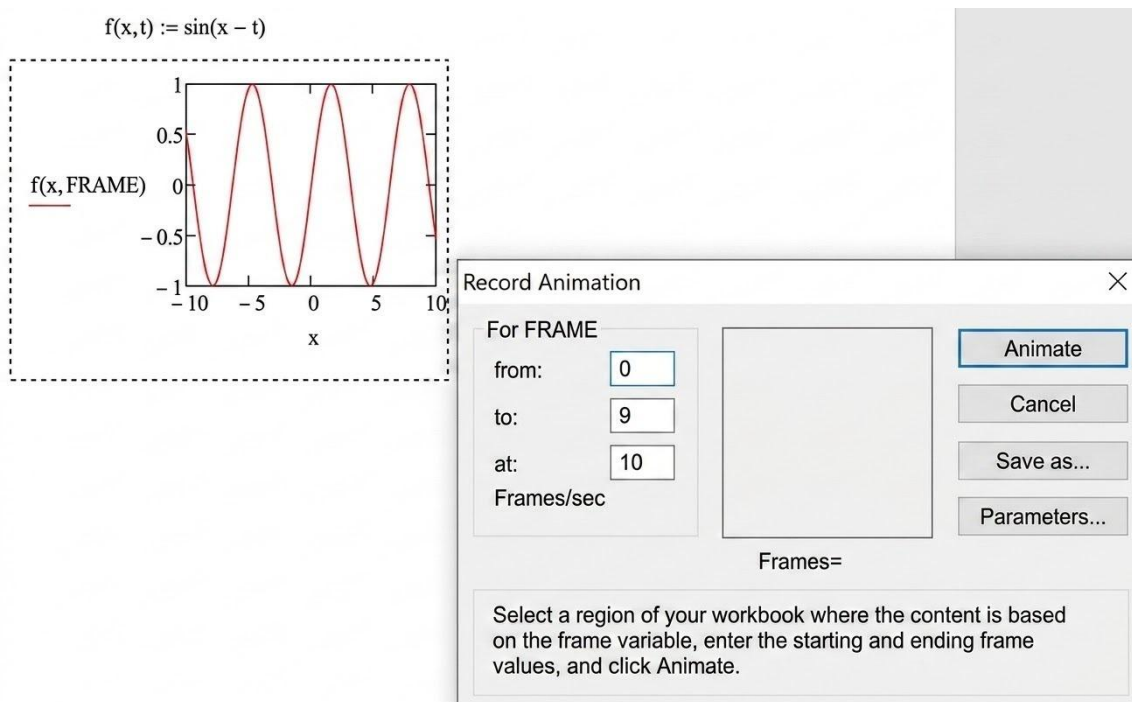


Рис. 1.59. Підготовка до створення ролика анімації

5. У діалоговому вікні **Запис анімації** натиснути кнопку **Анімувати** (Animate). Після цього у віконці діалогового вікна **Запис анімації** будуть з'являтися результати розрахунків виділеної області, що супроводжено виведенням поточного значення змінної **FRAME** (рис. 1.60). Після закінчення цього процесу на екрані з'явиться вікно програвача анімації (рис. 1.61).

6. Запустити перегляд анімації в програвачі, натискаючи кнопку відтворення в лівому нижньому кутку вікна програвача.

7. Якщо вид анімації влаштовує, зберегти її у вигляді відеофайлу, натиснувши кнопку **Зберегти як ...** (Save As) у діалоговому вікні **Запис анімації**. У діалоговому вікні **Зберегти анімацію** (Save Animation) звичайним для Windows способом вказати ім'я файлу і його розташування на диску.

8. Закрити діалог **Запис анімації** натисканням кнопки **Скасування** (Cancel) або кнопки управління його вікном.

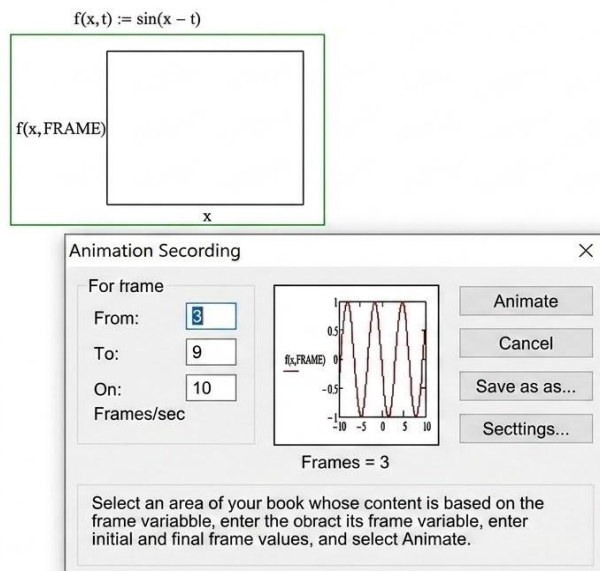


Рис. 1.60. Створення ролика анімації

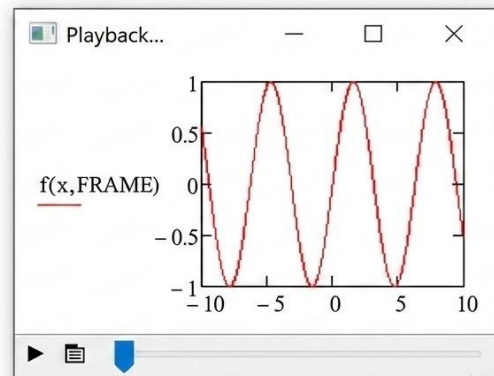


Рис. 1.61. Вікно програвача анімації

Збережений відеофайл можна використовувати за межами MathCAD. Отже, запускаючи відеофайли у звичайний спосіб, можна влаштувати барвисту презентацію результатів роботи як на своєму, так і іншому комп'ютері.

Для створення файлів анімації можна вибрати програму відеостискання (кодек) і якість компресії. Роблять це за допомогою кнопки **Параметри (Options)** діалогового вікна **Записування анімації**.

Примітка. Функцію $f(x,t) := \sin(x-t)$ для анімації можна записати не так, як на рис. 1.57, а так, як на рис. 1.62.

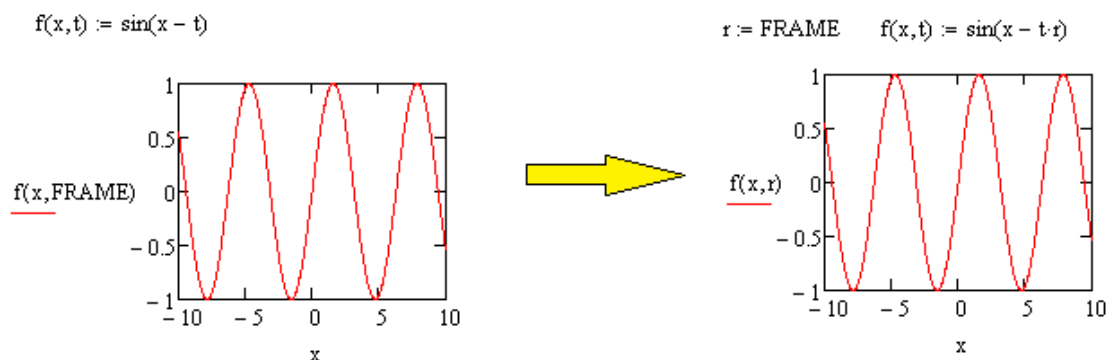


Рис. 1.62. Запис початкової функції в інший спосіб

1.6.2. Приклади створення анімаційного кліпу

Приклад 1. Розробити анімацію руху точки по синусоїді $y = \sin(t)$ на інтервалі від 0 до 9 із кроком 0,5. Аргумент t змінюється від 0 до 12 із кроком 0,1. Лістинг показано на рис. 1.63.

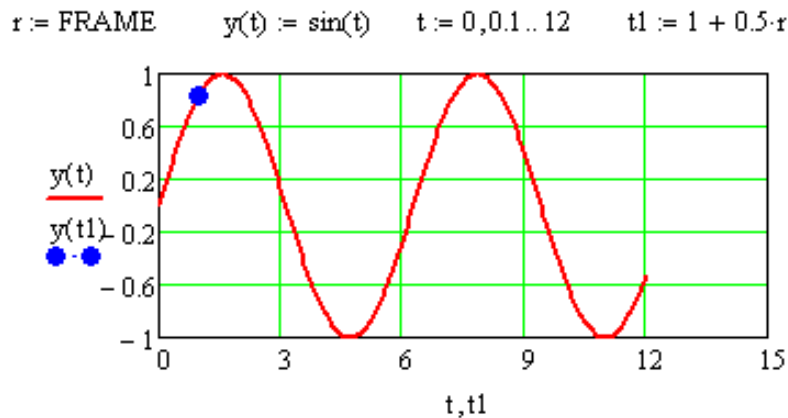


Рис. 1.63. Лістинг прикладу 1

Приклад 2. 3D-анімація. Змінювання поверхні $f(x, y) = a(x^2 + y^2)$ зі зміною параметра a від -2 до 2 із кроком 0,2. Сітка 20x20, інтервали аргументів $x(-2,2)$ і $y(-3,3)$. Лістинг показано на рис. 1.64.

Алгоритм виконання

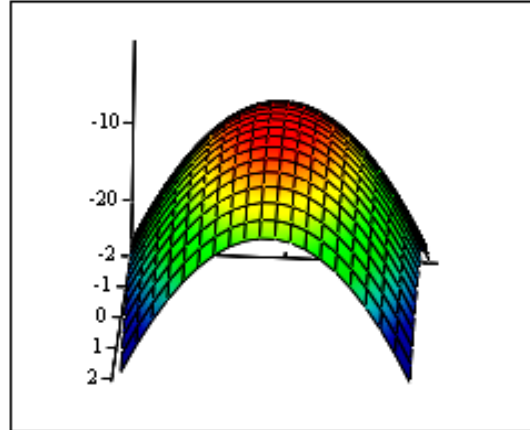
1. Виразити змінний параметр a через FRAME.
2. Ввести функцію.
3. Задавання граничних елементів і ліній сітки (можна задати безпосередньо у функції *CreateMesh*).
4. Створення матриці значень M .
5. Побудова поверхні (виклик шаблона графіка та введення матриці M).
6. Виклик режиму анімації.
7. Задавання області анімації та параметрів (кількість кадрів = $(a_{\max} - a_{\min})/\text{крок} = 4/0.2 = 20$ кадрів).
8. Створення ролика анімації.

```

a := -2 + 0.2·FRAME
f(x,y) := a·(x2 + y2)
x1 := -2    x2 := 2    y1 := -3    y2 := 3    nx := 20    ny := 20
M := CreateMesh(f,x1,x2,y1,y2,nx,ny)

```

a = -2



M

Рис. 1.64. 3D-анімація

ЗАВДАННЯ ДЛЯ ВИКОНАННЯ

Завдання 1. Виконати демонстраційний приклад моделювання процесу поширення гармонійної хвилі, що біжить, відповідно до наведених у теоретичних відомостях залежностей (рис. 1.57-1.62). Зробити висновки про швидкість поширення хвилі та характер її переміщення.

Завдання 2. Створити ролик анімації зміни графіка функції (варіанти в табл. 1.9 за номером у журналі). Значення змінній задати самостійно як ранжовану.

Завдання 3. Виконати приклад 1 із теоретичних відомостей (вихідні дані – рис. 1.63).

Завдання 4. Створити ролик анімації руху точки заданою траєкторією (варіанти з табл. 1.24 за номером у журналі). Значення змінній задати самостійно як ранжовану.

Завдання 5. Створити ролик анімації в полярних координатах (варіанти з табл. 1.25 за номером у журналі). Зміну аргументу задати самостійно.

Завдання 6. Виконати приклад 2 із теоретичних відомостей 3D-анімації (вихідні дані рис. 1.63).

Додаткове завдання. Створити анімаційний ролик, що моделює переміщення рухомого навантаження вздовж балки (спрощена модель рейки).

Постановка задачі: розглянути функцію вигину балки за дії рухомої сили:

$$y(x, t) = A \cdot \exp(-\alpha \cdot (x - vt)^2),$$

де x — координата вздовж балки (рейки);

t — час (задано через FRAME);

v — швидкість руху навантаження;

A — максимальний прогин;

α — параметр, що визначає ширину зони деформації.

Вихідні дані:

- довжина балки (рейки): $L = 25$ м, тобто просторова змінна: $x := 0, 0.1, \dots, 25$;
- амплітуда прогину: $A = 0.004$ м (4 мм);
- параметр ширини зони деформації: $\alpha = 0.8$;
- щоб ролик був видовищним, краще брати не реальну швидкість поїзда (20–30 м/с), а масштабовану для анімації: $v = 1$ м/кадр, тоді максимум зміщується рівномірно і чітко.

1.7. Програмування в системі MATHCAD

Ознайомлення із засобами програмування, які надає система MathCAD і набуття досвіду роботи з ними.

1.7.1. Програмування в пакеті MathCAD

Для вставлення програмного коду в документи *MathCAD* використовують спеціальну панель **Програмування** (Programming) (рис. 1.65). Більшість кнопок цієї панелі виконані у вигляді текстового подання операторів програмування, тому зміст їх зрозумілий [1-3].

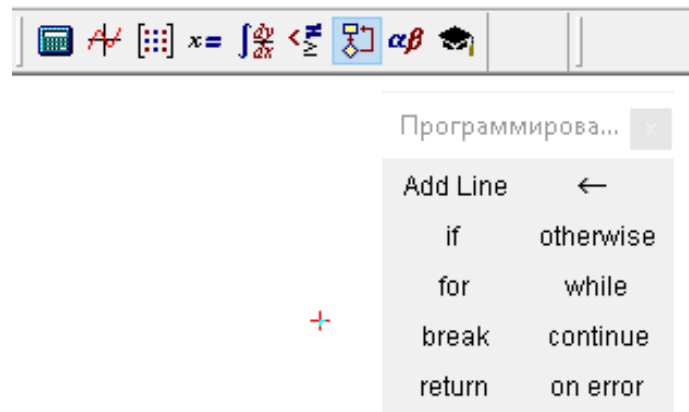


Рис. 1.65. Панель **Програмування**

Основними інструментами роботи в MathCAD є математичні вирази, змінні та функції. Призначення програмних модулів полягає у визначенні виразів, змінних і функцій у декілька рядків, дуже часто із застосуванням специфічних програмних операторів. Програмування в MathCAD має ряд переваг, які роблять документ більш простим і таким, що легко читати:

- можливість застосування циклів та умовних операторів;
- простота створення функцій і змінних;
- використання локальних змінних та обробка помилок [1-3].

Реалізувати алгоритм обчислення в пакеті MathCAD можна, використовуючи програми-функції, що містять конструкції, подібні до конструкцій мов програмування Pascal чи Fortran, оператори присвоєння, оператори циклів, умовні оператори і т. д. Такий спосіб програмування називають *програмуванням у програмі-функції*.

1.7.2. Використання програми-функції MathCAD

Опис програми-функції міститься в робочому документі перед викликом програми-функції:

- **ім'я** програми-функції. Кожна програма-функція *MathCAD* має своє ім'я, використовуючи яке, звераються до цієї програми-функції;
- **список формальних параметрів** (він може бути відсутнім). Після імені в круглих дужках записано *список формальних параметрів* (через кому), через які передають дані в програму-функцію для обчислень. Якщо програма-функція не має формальних параметрів, тоді дані передають через імена змінних, визначених вище опису програми-функції;
- **тіло програми-функції**. Тіло програми-функції містить локальні оператори присвоєння, умовні оператори, оператори циклу, а також інші програми-функції та функції користувача [1-3].

Порядок опису програми-функції MathCAD

Для введення в робочий документ опису програми-функції необхідно (рис. 1.66):

- ввести ім'я програми-функції, список формальних параметрів і символ «:»;
- відкрити панель *Програмування* та натиснути кнопку **Add line**.

На екрані з'явиться вертикальна риска і вертикальний стовпець із двома полями для введення операторів, що утворюють тіло програми-функції;

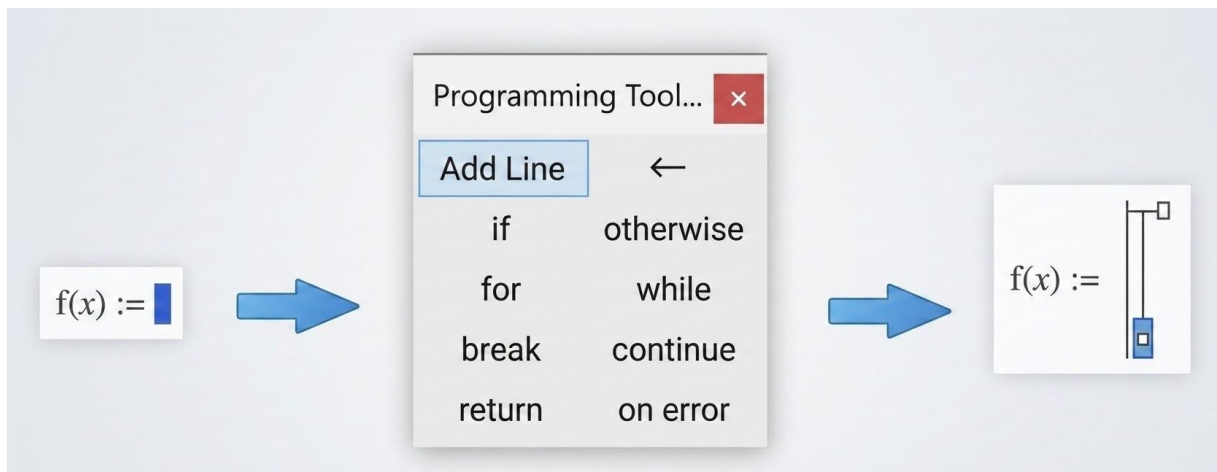


Рис. 1.66. Порядок опису програми функції

- перейти в перше поле і ввести перший оператор тіла програми-функції. Нижнє поле завжди призначене для визначення значень, які повертає програма. Для того щоб ввести додаткові поля для введення операторів, потрібно натиснути кнопку **Add line**. Для видалення того чи іншого оператора або поля введення з тіла програми-функції потрібно виділити його рамкою і натиснути клавішу **Delete**;
- заповнити нижнє поле введення, ввести вираз, який визначає значення, що повертається через ім'я програми-функції [1-3].

1.7.3. Оператори мови програмування в MathCAD

1.7.3.1. Локальний оператор присвоєння. Локальний оператор присвоєння використовують для задавання всередині програми значення змінної, він має такий вигляд:

$\langle \text{ім'я змінної} \rangle \leftarrow \langle \text{вираз} \rangle.$

Використання звичайного оператора присвоєння ($:=$) у тілі програми-функції призводить до синтаксичної помилки.

Приклад опису програми-функції

Обчислити значення функції $f(x) = \begin{cases} x^2 + 1 & \text{якщо } x \leq -1 \\ \sin(x) & \text{якщо } x > 0 \end{cases}$.

1. Опис і звертання до програми-функції, яка використовує формальні та фактичні параметри (рис. 1.67).

$$f(x) := \begin{cases} f \leftarrow x^2 + 1 & \text{if } x \leq -1 \\ f \leftarrow \sin(x) & \text{if } x > 0 \\ f & \end{cases} \quad f(-2) = 5 \quad f\left(\frac{\pi}{2}\right) = 1$$

Рис. 1.67. Використання функції з формальними і фактичними параметрами

2. Опис і звертання до програми-функції, яка не використовує формальні параметри (рис. 1.68).

$$x := -2 \quad f := \begin{cases} f \leftarrow x^2 + 1 & \text{if } x \leq -1 \\ f \leftarrow \sin(x) & \text{if } x > 0 \\ f & \end{cases} \quad f = 5$$

Рис. 1.68. Використання функції без формальних параметрів

1.7.3.2. Умовний оператор *if*. Цей оператор використовують тільки в тілі програми-функції, і для його введення потрібно натиснути кнопку **if** панелі програмування. На екрані з'явиться така конструкція:

$$f(x) := \begin{cases} \blacksquare & \text{if } \blacksquare \\ \blacksquare & \end{cases}$$

У поле, що знаходиться після *if*, вводять логічний вираз. У поле, що знаходиться перед *if*, вводять вираз, значення якого використано, якщо логічний вираз набуває значення *true*. У поле, що знаходиться в наступному рядку після *if*, вводять вираз, значення якого використано, якщо логічний вираз набуває значення *false*. Для введення в це поле необхідно (рис. 1.69):

- помістити це поле в рамку;
- натиснути кнопку *otherwise* панелі програмування;
- у поле, що залишилося, ввести відповідний вираз.

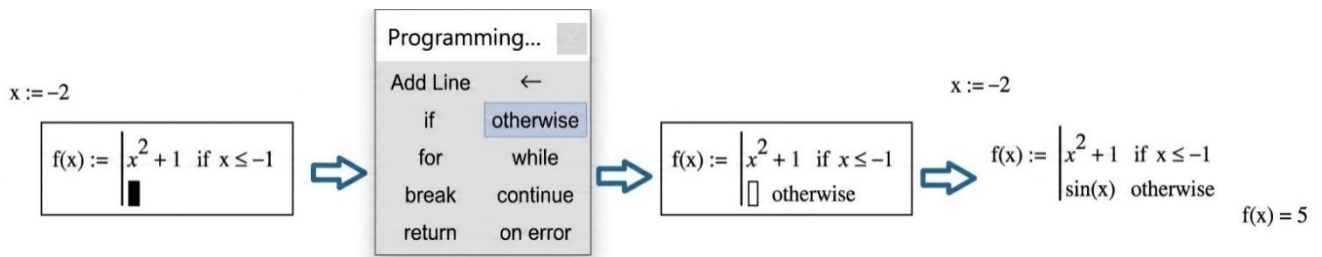


Рис. 1.69. Використання оператора *if*

1.7.3.3. Логічні операції. Логічну операцію **OR** позначають знаком + або $\vee$ і записують як

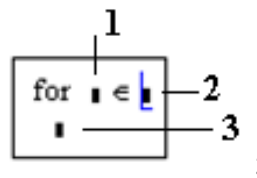
<логічний вираз 1> $\vee$ <логічний вираз 2>.

Логічну операцію **AND** позначають знаком * або $\wedge$ і записують як

<логічний вираз 1> $\wedge$ <логічний вираз 2>.

1.7.3.4. Оператор циклу з параметром. Для введення цього оператора необхідно виконати такі дії:

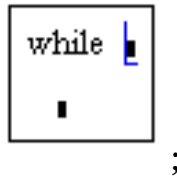
- натиснути кнопку **for** панелі *Програмування*. На екрані з'являться поля введення (місцезаповнювачі)



- у *поле 1* вводять ім'я параметра циклу;
- у *поле 2* вводять діапазон значень параметра циклу, використовуючи для цього дискретний аргумент;
- у *поле 3* вводять оператори, що складають тіло циклу.

1.7.3.5. Оператор циклу з передумовою. Для введення цього оператора необхідно виконати такі дії:

- натиснути кнопку **while** панелі *Програмування*. На екрані з'являться поля введення (місцезаповнювачі)



- у *поле 1* вводять умову виконання циклу;
- у *поле 2* вводять оператори тіла циклу. У тілі циклу мають бути присутні оператори, які роблять умову циклу хибною, інакше цикл буде виконуваний нескінченно.

Оператор циклу з передумовою виконуваний так: знайшовши оператор **while**, MathCAD перевіряє вказану умову. Якщо вона істинна, то виконуване тіло циклу, і знову перевіряють умову. Якщо вона хибна, то цикл завершується.

У системі *MathCAD* використовують масиви двох типів: одновимірні (вектори) і двовимірні (матриці). Порядковий номер елемента одновимірного масиву називають індексом. Індеси починаються з нуля або одиниці залежно від значення системної змінної **ORIGIN**. Вектори і матриці можна задавати різними способами:

- за допомогою команди *Додати* → *Матрицю*;
- допомогою комбінації клавіш **Ctrl+M**;
- натисканням кнопки  панелі *Математика*.

Щоб звернутися до окремих елементів вектора, використовують оператор нижнього індексу . Для роботи з масивами використовують вбудовані в *MathCAD* функції (табл. 1.26), які викликають командами *Додати* → *Функцію*.

Вбудовані функції

Функція	Призначення	Приклад
cols	Повертає кількість стовпців	cols(A)=1
rows	Повертає кількість рядків	rows(A)=5
last	Повертає індекс останнього елемента масиву	last(A)=5
length	Повертає кількість елементів масиву	length(A)=5
min	Повертає мінімальний елемент	min(A)=-1
max	Повертає максимальний елемент	max(A)=4

Обробка елементів двовимірного масиву відбувається так само, як і обробка елементів одновимірного масиву. Єдина відмінність – необхідно використовувати вкладені цикли: один цикл, зовнішній, для переміщення між рядками, а другий, внутрішній, для переходу між елементами рядка. Над двовимірними масивами визначені функції **cols**, **rows**, **min**, **max**.

ЗАВДАННЯ ДЛЯ ВИКОНАННЯ

Завдання 1. Розробити програму для обчислення значення функції (варіанти в табл. 1.27, потрібні дані, яких немає, задати самому).

Варіанти для завдання 1

Варіант	Функція
1	2
1	$y = \begin{cases} \sin^2 x, & \text{якщо } 0 \leq x < 1 \\ \sqrt{\ln x}, & \text{якщо } 1 \leq x \leq 3 \\ e^x, & \text{якщо } x > 3 \end{cases}$
2	$y = \begin{cases} 0, & \text{якщо } x < -1 \\ \operatorname{ctg}(x-1), & \text{якщо } -1 \leq x \leq 0 \\ \ln x, & \text{якщо } x > 0 \end{cases}$
3	$y = \begin{cases} 2^{x-1}, & \text{якщо } x < -8,6 \\ (x-1)^2, & \text{якщо } -8,6 \leq x \leq 8,6 \\ 2,7, & \text{якщо } x > 8,6 \end{cases}$

4	$y = \begin{cases} x^2 - e^{-x}, & \text{якщо } x < 2 \\ \ln x^2, & \text{якщо } x \leq -2 \\ \sin^2 x, & \text{якщо } x \geq 2 \end{cases}$
5	$y = \begin{cases} \ln(x^2 + 1), & \text{якщо } x \geq 5 \\ \sqrt[3]{9x + 2}, & \text{якщо } 3 \leq x < 5 \\ e^{x+0,5}, & \text{якщо } x < 3 \end{cases}$
6	$y = \begin{cases} \sqrt{ abx }, & \text{якщо } x \leq 8 \\ \ln(x - b)^2, & \text{якщо } x > 25 \\ e^{\sin ax}, & \text{якщо } x \leq 25 \end{cases}$
7	$y = \begin{cases} e^{-x}, & \text{якщо } x > 1 \\ \ln\sqrt{-x}, & \text{якщо } x \leq 1 \end{cases}$
8	$y = \begin{cases} \sqrt{x}, & \text{якщо } x > 0 \\ 2 - x^2, & \text{якщо } x \leq 0 \end{cases}$
9	$y = \begin{cases} \frac{\sin x}{x}, & \text{якщо } x > 0 \\ 2x^2 + \ln x , & \text{якщо } x \leq 0 \end{cases}$
10	$y = \begin{cases} 124 - e^x, & \text{якщо } x \leq 1 \\ tg(x - 1), & \text{якщо } 1 < x \leq 10 \end{cases}$

Завдання 2. Розробити програму для обчислення значення функції (варіанти в табл. 1.28). Три числа задати формулами: $a = \frac{n+m}{5}$, $b = \frac{n}{m+5}$, $c = m \cdot n + 24$. Значення m і n задати самостійно так, щоб серед чисел a , b , c були додатні і від'ємні.

Таблиця 1.28

Варіанти для завдання 2

Варіант	Функція	Варіант	Функція
1	2	3	4
1	Знайти мінімальне з чисел a, b, c	2	Знайти максимальне з чисел a, b, c
3	Знайти мінімальне число із a, b, c	4	Знайти максимальне число із a, b, c
5	Вибрати серед a, b, c від'ємні та надрукувати їх і квадратні корені з їхніх модулів	6	Вибрати серед a, b, c ті, що належать проміжку $[-4, 7]$

1	2	3	4
7	Вибрати серед a, b, c ті, модулі яких менше 5	8	Задані два вектори a і b . Знайти їхні максимальні значення та обчислити: $x = \max(a, b) - 1$
9	Знайти число $x = \min(a, b) + \max(ab, c, b)$	10	Знайти число $x = \frac{\min(a, c)}{\max(c, b)}$

Завдання 3. Розробити програму обчислення значення функції, починаючи із заданої точки (варіанти в табл. 1.29).

Таблиця 1.29

Варіанти для завдання 3

Варіант	Функція
1	$z = \frac{2\sin^2 x}{3 x + 1}, x \geq -2.5, \Delta x = 0.15, n = 6$
2	$z = \frac{e^{2x} - 8}{x + 3}, x \geq 1.5, \Delta x = 0.3, n = 5$
3	$z = \frac{x + \cos 2x}{3x}, x \geq 1.3, \Delta x = 0.2, n = 7$
4	$z = \frac{x + \cos 2x}{x + 2}, x \geq 0.6, \Delta x = 1.5, n = 6$
5	$z = \frac{\cos^2 x}{x^2 + 1}, x \geq 0.5, \Delta x = 0.1, n = 9$
6	$z = \frac{x^2 + 2x}{3\cos\sqrt{x} + 1}, x \geq 0.3, \Delta x = 0.9, n = 7$
7	$z = \frac{x + \sin 2x}{x^2 + 2}, x \geq 3.1, \Delta x = 0.8, n = 6$
8	$z = \frac{x^2 - 2}{3 \ln x}, x \geq 2, \Delta x = 1.5, n = 5$
9	$z = \frac{2,3x + 8}{ 2 \cos x + 1}, x \geq 0.4, \Delta x = 0.9, n = 7$
10	$z = \frac{\arccos x}{2x + 1}, x \geq 0, \Delta x = 0.2, n = 4$

Завдання 4. Розробити програму обчислення значення суми та добутку (варіанти в табл. 1.30).

Таблиця 1.30

Варіанти для завдання 4

Варіант	Функція	
1	$s = \sum_{i=1}^{10} \frac{i^2 + 1}{i^3 + 2}$	$f = \prod_{k=m}^n \frac{k^{m-1}}{(k+1)(k^2-1)} \quad m = 5$ $n = 13$
2	$y = \sum_{i=k}^n \frac{i+1}{(i+3)(i+6)} \quad k = 1 \quad n = 9$	$p = \prod_{i=8}^{17} (-1)^i \frac{i^2 - 2}{i+3}$
3	$s = \sum_{k=6}^{18} \frac{k^4 + 2}{2k^2 - 1}$	$f = \prod_{k=m}^j \frac{k+3}{k^2 + 10k + 24} \quad m = 3$ $n = 13$
4	$y = \sum_{j=n}^i \frac{j+2}{(j+4)(j+5)} \quad i = 12 \quad n = 3$	$p = \prod_{s=5}^{11} (1)^s \frac{s^2 + 3}{3s + 2}$
5	$s = \sum_{k=5}^{15} (-1)^k \frac{k^4 - 2}{3k + 1}$	$f = \prod_{n=m}^i \frac{n^{i-m} + 1}{n + 3} \quad m = 4 \quad i = 12$
6	$y = \sum_{n=i}^k \frac{n-2}{n^2 + 16} \quad i = 2 \quad k = 7$	$p = \prod_{k=2}^{21} (-1)^{k+1} \frac{k+3}{3k+1}$
7	$s = \sum_{i=2}^{12} (-1)^i \frac{i^4 - 2}{i^2 + 3}$	$f = \prod_{k=i}^j \frac{(k+3)^{j-k}}{(k+7)(k+9)} \quad i = 5$ $j = 11$
8	$s = \sum_{k=2}^{32} (-1)^{-k} \frac{(k-1)(k-2)}{k+3}$	$f = \prod_{n=l}^m \frac{(n+3)^2 + 1}{(n-5)^3} \quad m = 14 \quad l = 6$
9	$y = \sum_{n=i}^k \frac{n}{n^2 + 5n + 6} \quad i = 4 \quad k = 11$	$p = \prod_{i=12}^{19} (-1)^{i-1} \frac{3i-5}{i^2 + 8}$
10	$y = \sum_{n=k}^i \frac{(n+3)^2}{(n+5)(n+7)} \quad i = 15 \quad k = 5$	$d = \prod_{l=8}^{17} (-1)^{l-1} \frac{l^{2+l} - 2}{l+3}$

Завдання 5. Розробити програму для завдань за варіантами з табл. 1.31.
Значення елементів масиву задати самостійно (значення від'ємні і додатні).

Таблиця 1.31

Варіанти для завдання 5

Варіант	Завдання
1	Визначити порядкові номери додатних елементів масиву A(9)
2	Обчислити середнє арифметичне елементів масиву C(9), значення яких більше 5
3	Визначити, який номер має найменший елемент масиву B(8)
4	Знайти суму значень додатних елементів масиву B(6)
5	Знайти кількість додатних елементів масиву C(8)
6	Знайти середнє арифметичне від'ємних елементів масиву
7	Знайти кількість елементів масиву Y(7), значення яких менше 0.99
8	Знайти добуток значень елементів масиву, значення яких більше 2.0
9	Знайти суму значень від'ємних елементів масиву C(9)
10	Знайти максимальний елемент масиву B(7)

Завдання 6. Розробити програму для завдань за варіантами з табл. 1.32.

Таблиця 1.32

Варіанти для завдання 6

Варіант	Завдання	Варіант	Завдання
1	2	3	4
1	Усі від'ємні елементи матриці замінити на 25 і вивести на екран створену матрицю. $B(3,4) = \begin{pmatrix} 1.2 & -2.4 & 2.7 & -8.5 \\ 0.0 & 6.5 & -8.1 & -1.5 \\ 1.3 & -8.4 & 7.8 & 4.9 \end{pmatrix}$	2	Замінити нулями всі додатні елементи матриці і вивести на екран нову матрицю. $B(3,4) = \begin{pmatrix} 0.0 & 6.5 & -8.1 & -4.5 \\ 8.5 & 7.8 & -3.4 & 6.1 \\ 3.3 & -8.4 & 7.8 & 4.9 \end{pmatrix}$
3	Замінити всі нулі матриці числом 34 і вивести на екран нову матрицю. $C(2,4) = \begin{pmatrix} 7 & 0 & -6 & 4 \\ 0 & 4 & 3 & 0 \end{pmatrix}$	4	Усі додатні елементи матриці збільшити на 10 і вивести на екран цю матрицю і створену матрицю. $X(3,2) = \begin{pmatrix} 5 & 0 \\ -4 & 3 \\ 5 & 1 \end{pmatrix}.$

Продовження табл. 1.32

1	2	3	4
5	Вивести на екран середнє арифметичне всіх елементів матриці і саму матрицю А. $A(3,3) = \begin{pmatrix} 5 & 0 & -6 \\ 4 & 3 & 2 \\ 4 & 8 & 3 \end{pmatrix}$	6	Знайти квадрат найбільшого елемента масиву. $H(4,4) = \begin{pmatrix} 3 & 0 & 0.1 & 2.5 \\ -1 & 2 & -5 & 1.3 \\ 66 & -2.5 & 0.9 & 0.5 \\ 11 & 2.5 & 0 & 1 \end{pmatrix}$
7	У матриці поміняти місцями перший і другий стовпці. $D(4,4) = \begin{pmatrix} 4 & 5 & -0.5 & -2.5 \\ 1 & 2 & 5 & 1.3 \\ 3.8 & 0 & 0.1 & 0.5 \\ 1 & 0.9 & 2 & 3.4 \end{pmatrix}$	8	У матриці знайти суму елементів, що лежать вище головної діагоналі. $H(4,4) = \begin{pmatrix} 3 & 4 & -0.1 & 2.5 \\ -1 & 2 & 5 & 1.3 \\ 3 & 0 & 0.1 & 0.5 \\ 1 & 0 & 2 & 3.4 \end{pmatrix}$

Додаткове завдання Розробити програму для завдань за варіантами з табл. 1.33.

Таблиця 1.33

Варіанти для додаткового завдання

Варіант	Завдання	Варіант	Завдання
1	2	3	4
1	Визначити кількість додатних і від'ємних елементів матриці $A(10, 15)$	2	Обчислити суми елементів кожного рядка матриці $X(20,20)$, визначити найменше значення цих сум і номер відповідного рядка
3	Нормувати квадратну матрицю, поділивши всі елементи кожного рядка на найбільший за модулем елемент цього рядка	4	У прямокутній матриці переставити стовпці так, щоб сума елементів стовпця зростала від першого стовпця до останнього
5	Знайти найменший елемент матриці $X(15, 20)$ і записати нулі в той рядок і стовпець, де він знаходиться	6	Дано два одновимірні масиви довільної розмірності. Знайти суму максимальних елементів кожного масиву

Продовження табл. 1.33

1	2	3	4
7	Обчислити суму всіх непарних елементів матриці $D(30,25)$, які розміщені в парних рядках	8	Є дві матриці. Підрахувати кількість елементів, які більше за абсолютним значенням, ніж число S (задається довільно)
9	Обчислити суму елементів матриці $A(20, 20)$, розміщених над головною діагоналлю	10	Знайти найбільші елементи кожного рядка матриці $X(20, 20)$ і записати їх у масив Y

Розділ 2

МАТРИЧНА ЛАБОРАТОРІЯ MATLAB

2.1. MATLAB у математичних розрахунках. Обчислення математичних виразів

Ознайомлення з візуальним середовищем математичного моделювання MATLAB; вивчення інтерфейсу користувача системи MATLAB і основ роботи з системою в режимі прямих обчислень.

2.1.1. Робоче середовище MATLAB

MATLAB – це математичний пакет прикладних програм, заснований на використанні матриць. MATLAB містить велику кількість спеціалізованих програм (функцій), має власну мову програмування високого рівня, а також надає потужні можливості візуального подання двовимірних і тривимірних даних. Важливу роль у MATLAB відіграють спеціалізовані групи програм (пакети) – так звані Toolbox, у яких зібрані функції для розв’язання окремих класів задач, наприклад PDE Toolbox, Spline Toolbox та ін. [4-7].

Робоче середовище MATLAB (рис. 2.1) включає головне меню **1**, панель інструментів **2**, а також п’ять спеціалізованих вікон:

- **командне вікно 3 (Command Window)** містить командний рядок, який розпочинається зі знака запрошення `>>` і призначений для введення користувачем математичних виразів, команд, а також виклику функцій. Введення кожної команди, виразу або функції має завершуватися натисненням клавіші `<Enter>`, у результаті чого в командному вікні нижче введеної команди буде виведений результат її виконання або інші повідомлення системи (наприклад повідомлення про помилку). Якщо немає потреби одразу після виконання команди виводити її результат на екран, то в кінці команди в командному рядку треба поставити крапку з комою (;);

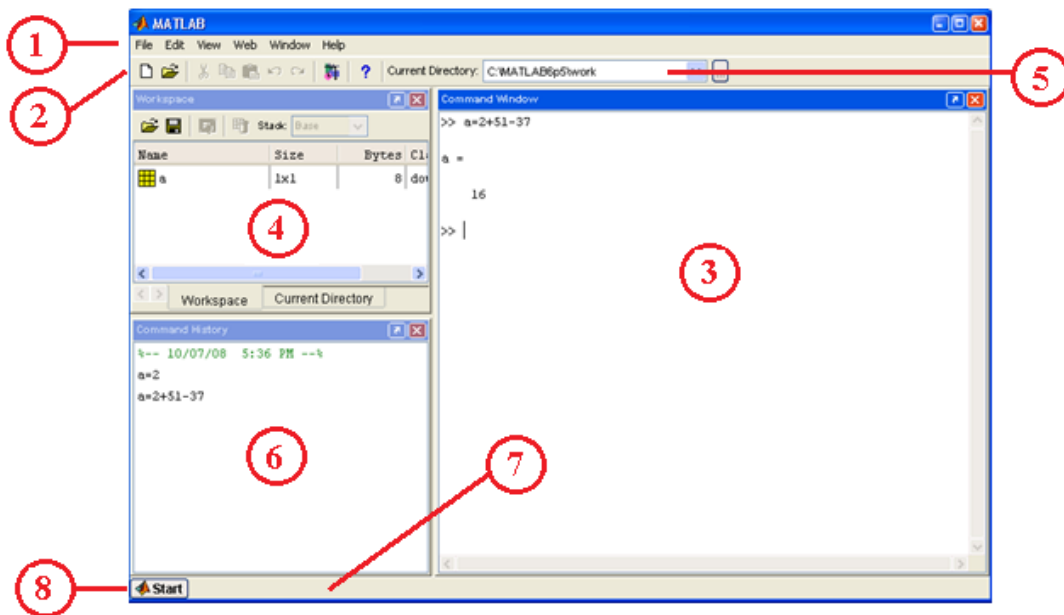


Рис. 2.1. Інтерфейс програми MATLAB

- **вікно робочої області 4 (Workspace)** відображає всі змінні, що були створені в процесі роботи з поточною сесією MATLAB;

- **вікно поточного каталогу 5 (Current Directory)** дає змогу змінити поточний каталог і переглянути його вміст; усі файли, що зберігає або відкриває користувач, за замовчуванням містяться в поточному каталозі, яким на початку є підкаталог work основного каталогу MATLAB;

- **вікно історії команд 6 (Command History)** зберігає всі команди, які були введені в командному рядку; отже, команду, яка вже виконувана раніше, можна віднайти в історії команд і повторити її;

- **вікно панелі запуску 7 (Launch Pad)** надає ще один спосіб доступу до встановлених на комп'ютері прикладних пакетів, що входять до складу MATLAB; аналогічний доступ можна отримати, натиснувши кнопку **Start** (8), розташовану в лівому нижньому куті вікна MATLAB [4-7].

Слід звернути увагу на такі кнопки панелі інструментів:

New M-file - виводить порожнє вікно редактора m-файлів;

Open file - відкриває вікно для завантаження файлів MATLAB;

Simulink - відкриває вікно браузера бібліотек Simulink;

Help - відкриває вікно довідки.

Ці функції дубльовані в дуже простому меню системи MATLAB.

Уся робота організована через командне вікно (**Command Window**), яке з'являється із запуском програми. У процесі роботи дані розташовані в пам'яті (**Workspace**) у вигляді матриць [4-7].

2.1.2. Арифметичні операції

Система MATLAB працює як із дійсними, так і комплексними числами. Перед використанням операцій із комплексними числами необхідно визначити змінну $i = \text{sqrt}(-1)$ або $j = \text{sqrt}(-1)$.

У середовищі MATLAB визначені такі арифметичні операції:

- 1) $\wedge$ - піднесення до степеня;
- 2) $*$ - множення,
 $/$ - ділення зліва направо,
 $\backslash$ - ділення справа наліво;
- 3) $+$ - додавання,
 $-$ - віднімання.

Тут операції наведені в порядку спадання пріоритету їх виконання для обчислення виразів. Для зміни порядку виконання арифметичних операторів необхідно використовувати круглі дужки ().

Використання MATLAB у режимі калькулятора може відбуватися простим записом у командний рядок послідовності арифметичних дій із числами [4-7].

Наприклад, для звичайного арифметичного виразу $-4,5^2 \cdot 7,23 - 3,14 \cdot 10,4$ у MATLAB потрібно записати

$$(-4.5)^2 * 7.23 - 3.14 * 10.4.$$

Після введення з клавіатури цієї послідовності і натискання клавіші <Enter> у командному вікні буде виведений результат:

```
>> (-4.5)^2*7.23 - 3.14*10.4
```

```
ans =
```

```
113.7515
```

Арифметичні вирази в MATLAB зазвичай сформовані з операторів, констант (або чисел), змінних і функцій [4-7].

2.1.3. Змінні та константи

2.1.3.1. Числа в MATLAB можуть бути цілими чи дробовими, дійсними (із фіксованою або рухомою крапкою) чи комплексними, додатними чи від'ємними. Наприклад, 25 – ціле число; $2/3$ – дробове число ($\frac{2}{3}$); 0.17 – дійсне число з фіксованою крапкою (0,17); $1.7e-1$ – дійсне число з рухомою крапкою ($1,7 \cdot 10^{-1}$); $4.8+0.6i$ – комплексне число ($4,8 + 0,6 \cdot i$); 9.3 (або +9.3) – додатне число; -6.57 – від'ємне число.

У середовищі MATLAB визначені декілька спеціальних **констант** – сталих чисельних значень, що позначені унікальним ідентифікатором і мають математичний смисл, які можна використовувати для обчислення виразів (табл. 2.1), і спеціальних символів (табл. 2.2) [4-7].

Зазначимо, що ім'я NaN (Not-a-Number) зарезервовано для результату операцій $0/0$, $0*\text{inf}$, $\text{inf}-\text{inf}$ тощо [4-7].

Таблиця 2.1

Зарезервовані імена констант

Ім'я	Опис
1	2
ans	Результат останньої операції
i, j	Уявна одиниця
pi	Число $\pi \approx 3,14159265358979$
eps	Машинна точність, із якою за замовчуванням виконувані обчислення арифметичних операцій і більшості вбудованих функцій

1	2
realmax	Максимальна дійсне число
realmin	Мінімальна дійсне число
inf	Нескінченність ∞
NaN	Дані нечислового типу (Not a Number)
end	Найбільше значення індексу розмірності масиву

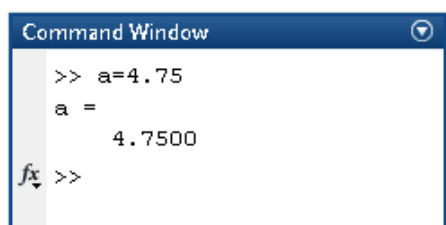
Таблиця 2.2

Спеціальні символи

Символ	Призначення
[]	Квадратні дужки використовують для завдання матриць і векторів
	Пробіл служить для поділу елементів матриць
,	Кома застосовувана для поділу елементів матриць і оператора в рядку введення
;	Крапка з комою відокремлює рядки матриць, а крапка з комою наприкінці оператора (команди) скасовує виведення результату на екран
:	Двокрапка використовується для зазначення діапазону (інтервалу зміни величини) і як знак групової операції над елементами матриць
()	Круглі дужки застосовують для задавання порядку виконання математичних операцій, а також зазначення аргументів функцій і індексів матриць
.	Точка відокремлює дробову частину числа від цілої його частини, а також застосовувана в складі комбінованих знаків (. *,. ^,./,. \)
...	Три крапки і більше наприкінці рядка указують на продовження виразу в наступному рядку
%	Знак відсотка позначає початок коментаря
'	Апостроф вказує на символні рядки, а для включення самого апострофа в символний рядок потрібно поставити два апострофи поспіль

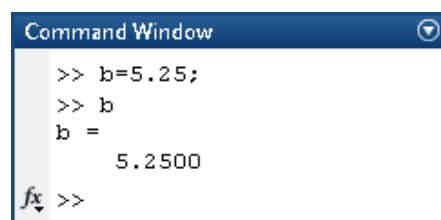
2.1.3.2. Змінні в MATLAB характеризує ім'я, яке також називають ідентифікатором, і значення. У MATLAB кількість символів в імені змінної обмежена і дорівнює 31. Ідентифікатор має починатися з літери та може містити літери, цифри, деякі інші допустимі символи (окрім пробілу). Великі та малі літери в іменах змінних, функцій і команд розрізняються, наприклад у і Y є двома різними змінними [4-7].

Щоб присвоїти змінній деяке значення, використовують знак **=** (*оператор присвоювання*). Формат присвоювання: **>> ім'я змінної = вираз**. Після натискання клавіші <Enter> система автоматично виведе результат на екран (рис. 2.2, а).



```
>> a=4.75
a =
    4.7500
fx >>
```

а



```
>> b=5.25;
>> b
b =
    5.2500
fx >>
```

б

Рис. 2.2. Оператор присвоювання

Якщо немає потреби одразу після виконання команди виводити її результат на екран, то наприкінці команди в командному рядку треба поставити крапку з комою (;). У подальшому переглянути значення змінної *b* можна у вікні робочої області (Workspace), або ввівши в командному рядку її ім'я **>> b** і натиснувши клавішу <Enter> (рис. 2.2, б).

Слід зазначити, що в одному командному рядку можуть бути записані одразу декілька команд, розділених комою (,) або крапкою з комою (;). Введення команд через кому еквівалентне їх послідовному виконанню в командному рядку з відображенням результатів на екрані, тоді як із записом їх через крапку з комою виведення результатів не відбувається.

Зазначимо, що за відсутності імені змінної в лівій частині та знака = автоматично створюється змінна **ans** (скорочено від англ. «answer»), якій присвоєно результат обчислення виразу [4-7].

Приклад 1. Обчислити в MATLAB значення виразу $4^3 + 7x \cdot (5,2 - \frac{1}{3})$, якщо $x = 1.03$ (рис. 2.3).

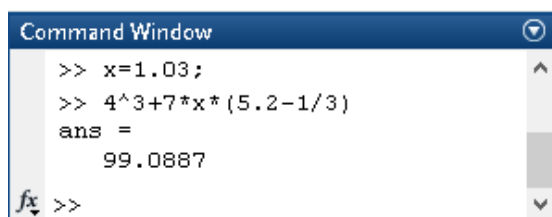


Рис. 2.3. Вигляд екрана прикладу 1

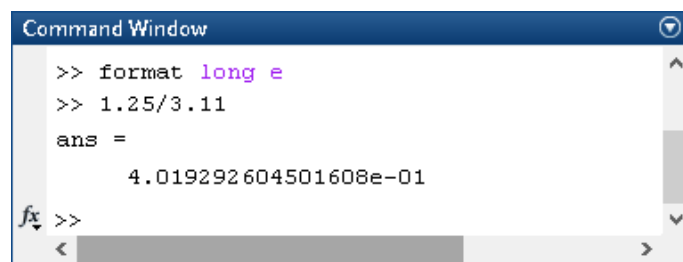
За замовчуванням результати обчислень виведені у командне вікно з чотирма знаками після десяткової точки. Усі розрахунки в MATLAB виконують із подвійною точністю, а для подання чисел на екрані є різні формати. Потрібний формат може бути визначений у меню **File/Preferences** або за допомогою команди **format** (табл. 2.3) [4-7].

Таблиця 2.3

Формати подання даних у пакеті MATLAB

Команда	Опис формату подання числа
format short	із фіксованою точкою та чотирма знаками після точки (за замовчуванням)
format long	із фіксованою крапкою та 14 знаками після крапки
format short e	із рухомою крапкою та чотирма десятковими знаками
format long e	із рухомою крапкою та 15 десятковими знаками
format short g	із фіксованою крапкою і трьома знаками після крапки
format long g	із фіксованою крапкою і 13 знаками після крапки
format rat	у вигляді звичайних дробів
format hex	у шістнадцятковій системі числення

Після виконання однієї з наведених команд встановлений формат буде збережений до наступної команди зі зміни формату (рис. 2.4).



```
>> format long e
>> 1.25/3.11
ans =
    4.019292604501608e-01
fx >>
```

Рис. 2.4. Виконання команди format

Змінні в MATLAB не потрібно попередньо описувати, вказуючи їхній тип. Усі дані зберігають у вигляді масивів: числові змінні (внутрішній тип numeric), текстові рядки (char), комірки (Cell) і структури (struct). *Двовимірний масив - це матриця, одновимірний - вектор, а скаляр - матриця розміру 1x1.* Ім'я змінної має починатися з літери, після неї можуть бути літери, цифри і символ підкреслення. Допустимі імена будь-якої довжини, але MATLAB ідентифікує їх за першими 31 символами і розрізняє великі і малі літери [4-7].

У командному вікні в режимі діалогу відбувається обчислення. Користувач вводить команди або запускає на виконання файли з текстами мовою MATLAB. Інтерпретатор обробляє введені значення і видає результати: числові і рядкові дані, попередження і повідомлення про помилки.

Рядок введення позначений знаком >>.

Для роботи з MATLAB в командному режимі діє найпростіший рядковий редактор. Зверніть особливу увагу на застосування клавіш **Page Up** і **Page Down** (стрілки курсору «Вгору» - ↑ і «Вниз» - ↓). Вони використовуються для підстановки після маркера рядка введення >> раніше введених команд, наприклад, для їх виправлення, дублювання або доповнення. При цьому зазначені клавіші забезпечують перегортання раніше введених рядків знизу вгору або зверху вниз.

Усі значення змінних, обчислені протягом поточного сеансу роботи, зберігають у спеціально зарезервованій області пам'яті комп'ютера, яку називають робочим простором системи MATLAB (**Workspace**) [4-7].

Для перегляду значення будь-якої змінної з поточного робочого простору системи достатньо набрати її ім'я і натиснути клавішу Enter.

Після закінчення сеансу роботи з системою MATLAB усі раніше обчислені змінні будуть видалені. Щоб зберегти у файлі на диску комп'ютера вміст робочого простору системи MATLAB, потрібно виконати команду меню **File \ Save Workspace As...** За принципом замовчування розширення імені файлу **mat**, тому такі файли прийнято називати *М-файлами*.

2.1.4. Алгебраїчні та тригонометричні функції

Система MATLAB дає змогу обчислювати різні математичні функції. Наступні елементарні алгебраїчні функції як аргумент використовують одне або два дійсних (x , y) або одне комплексне (z) число (табл. 2.4) [4-7].

Таблиця 2.4

Елементарні алгебраїчні функції

Функція	Опис
abs(z), abs(x),	Обчислення модуля числа
angle(z)	Обчислення аргументу z
sqrt(z), sqrt(x)	Обчислення квадратного кореня чисел z і x
real(z)	Обчислення дійсної частини комплексного числа z
imag(z)	Обчислення уявної частини комплексного числа z
round(x)	Округлення до цілого
fix(x)	Округлення до найближчого цілого в бік нуля
rem(x, y)	Обчислення залишку від ділення x на y
exp(x)	Обчислення e в степені x
log(x)	Обчислення натурального логарифма числа x
log10(x)	Обчислення десяткового логарифма числа x

Система MATLAB дає змогу для обчислення тригонометричних і зворотних тригонометричних функцій змінної x (табл. 2.5).

Таблиця 2.5

Тригонометричних функцій

Функція	Опис
$\sin(x)$	Обчислення синуса
$\cos(x)$	Обчислення косинуса
$\tan(x)$	Обчислення тангенса
$\text{asin}(x)$	Обчислення арксинуса
$\text{acos}(x)$	Обчислення арккосинуса
$\text{atan}(x)$	Обчислення арктангенса
$\text{atan2}(y, x)$	Обчислення арктангенса за координатами точки

Для обчислення математичного виразу потрібно набрати його відповідно до правил MATLAB.

Приклад 2. Обчислити значення виразу

$$e^{-2,5}(\ln(11,3))^{0,3} - \sqrt{\frac{\sin 2,45\pi + \cos 3,78\pi}{\text{tg } 3,3}}.$$

Результати показані на рис. 2.5.

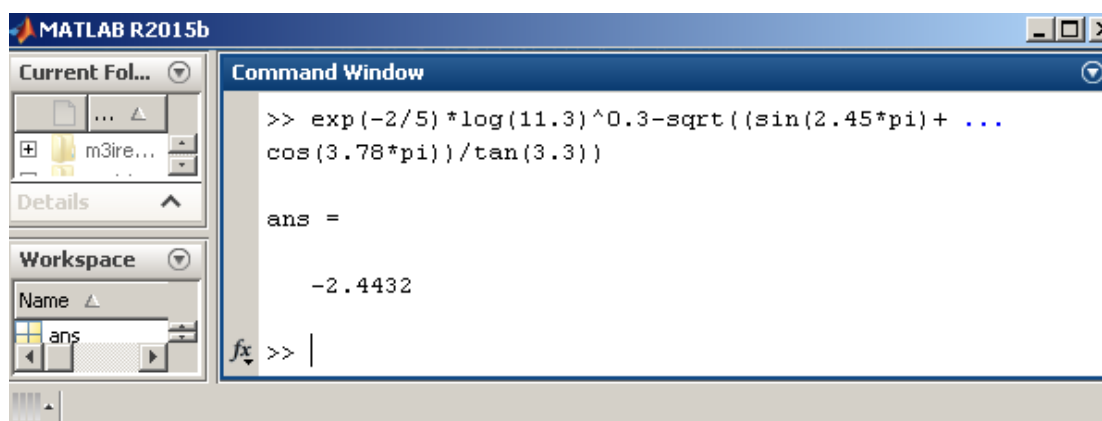


Рис. 2.5. Вигляд екрана прикладу 2

Для запису довгого виразу для перенесення частини його на інший рядок використовують *три крапки* (...) після попереднього пробілу.

Натискання клавіші *Enter* переводить курсор на новий рядок. Після завершення введення виразу натисканням клавіші *Enter* відповідь буде виведена в командне вікно (рис. 2.5) [4-7].

ЗАВДАННЯ ДЛЯ ВИКОНАННЯ

Завдання 1. Робота з деякими командами управління вікнами командного режиму.

1. Для отримання довідки про будь-який об'єкт використовують команди:

>> help <ім'я об'єкта>

або

>> doc <ім'я об'єкта>.

Самостійно отримати інформацію про арифметичні оператори **plus**, **minus**, формат виведення чисел **format**.

2. **clc** - очищає екран і курсор поміщає в лівому верхньому кутку екрана.

Очистіть екран.

Завдання 2. Самостійно обчислити:

- суму і різницю будь-яких цілих і дійсних двозначних чисел без присвоювання значень змінним;

- множення і частку від ділення будь-яких двох чисел із присвоюванням значень змінним;

- піднесення до степеня двох чисел;

- квадратний корінь із заданого числа з присвоюванням значення змінній і вивести її значення на екран;

- вираз $\sqrt[6]{128}$ і $\sqrt[6]{-128}$;

- перевірити результати обчислень 1/0 і 0/0.

Завдання 3. Встановлення формату виведення чисел. Застосуйте різні формати для виконання арифметичних операцій над двома числами.

Завдання 4. Обчислити вказаний арифметичний вираз за власним варіантом завдання (табл. 2.6).

Таблиця 2.6

Варіанти для завдання 4

Варіант	Арифметичний вираз	Варіант	Арифметичний вираз
1	$\frac{\left(12\frac{1}{6} - 6\frac{1}{27} - 5,25\right) \cdot 13,5 + 0,111}{0,02}$	2	$\frac{\left(6,6 - 3\frac{3}{14}\right) \cdot 5\frac{5}{6}}{(21 - 1,25) : 2,5}$
3	$\frac{\left(1\frac{1}{12} + 2\frac{5}{32} + \frac{1}{24}\right) : 9,6 + 2,13}{0,0004}$	4	$\frac{2,625 - \frac{2}{3} \cdot 2\frac{5}{14}}{\left(3\frac{1}{12} + 4,375\right) : 19\frac{8}{9}}$
5	$\frac{0,134 + 0,05}{18\frac{1}{6} - 1\frac{11}{14} - \frac{2}{15} \cdot 2\frac{6}{7}}$	6	$\frac{\left(58\frac{4}{15} - 56\frac{7}{24}\right) : 0,8 + 2\frac{1}{9} \cdot 0,225}{8,75 \cdot 0,6}$
7	$\frac{1\frac{4}{11} \cdot 0,22 : 0,3 - 0,96}{\left(0,2 - \frac{3}{40}\right) \cdot 1,6}$	8	$\frac{\left(\frac{0,216}{0,15} + 0,56\right) : 0,5}{\left(7,7 : 24,75 + \frac{2}{15}\right) \cdot 4,5}$
9	$\frac{\left(\frac{3}{5} + 0,425 - 0,005\right) : 0,12}{30,5 + \frac{1}{6} + 3\frac{1}{3}}$	10	$\frac{3\frac{1}{3} + 2,5}{2,5 - 1\frac{1}{3}} \cdot \frac{4,6 - 2\frac{1}{3}}{4,6 + 2\frac{1}{3}} : \left(\frac{0,05}{\frac{1}{7} - 0,125} + 5,7\right)$

Завдання 5. Обчислити математичний вираз для двох (*a* і *б*) випадків заданих значень параметрів за власним варіантом. У табл. 2.7 наведені варіанти завдань (у таблиці, *наприклад*, 18° - це 18 градусів).

Таблиця 2.7

Математичний вираз із відповідними параметрами для завдання 5

Варіант	Математичний вираз	Випадок <i>a</i>		Випадок <i>б</i>	
		<i>m</i>	<i>n</i>	<i>m</i>	<i>n</i>
1	2	3	4	5	6
1	$3m^2 + \sqrt[3]{2n^2} : m$	$-\frac{14}{5}$	$\operatorname{tg} \frac{\pi}{8}$	$2,2 \cdot 10^{-2}$	$\frac{1}{3,1}$
2	$\frac{4}{3} m^3 \sin^2 \frac{n}{2} \cdot \sqrt{\cos n}$	$1,7 \cdot 10^3$	18°	$\frac{16}{21}$	$\frac{\pi}{5}$

1	2	3	4	5	6
3	$\sqrt[3]{\frac{m\sqrt{n}}{tg61^\circ}}$	1,5	0,8	$3 \cdot 10^{-2}$	0,71
4	$\frac{3m^2\sqrt{6,8 \cdot (m-n)}}{4(m+n)^3}$	$4,13 \cdot 10^{-1}$	$\frac{1}{261}$	$\sin \frac{5\pi}{8}$	$-tg12^\circ$
5	$\frac{m^3}{6} \cos \frac{n}{2} \sqrt{\sin n}$	$\lg 2,38$	$\frac{\pi}{5}$	$e^{-0,3}$	65°
6	$\sqrt{\frac{m^3}{16,3 \cdot \sin n \cdot \sin 2n}}$	$3,15 \cdot 10^{-2}$	5°	$e^{3,5}$	$\frac{2\pi}{13}$
7	$5 \sin 35^\circ \sqrt{\frac{m^3 \cos 36^\circ}{\pi^3 tg(n)}}$	$\ln 3$	44°	$\frac{18}{25}$	$\frac{7}{12}\pi$
8	$ \lg(1 + \sin m) + \ln(1 - \sin n) $	$\frac{3\pi}{7}$	83°	$\frac{2\pi}{3}$	16°
9	$\sqrt[3]{\sin^2(m+n) - \sin^2(m-n)}$	$\frac{5\pi}{7}$	$0,3\pi$	12°	220°
10	$3\left(m^{-\frac{2}{3}} + n^{-\frac{1}{2}}\right) \cdot \sqrt[3]{m \cdot n}$	$\ln 3$	$\lg 3$	0,013	$1,4 \cdot 10^2$

Завдання 6. У командному вікні записати вираз мовою MATLAB та обчислити значення змінних відповідно до варіанта завдання (табл. 2.8).

Таблиця 2.8

Варіанти завдань

Варіант	Вираз	Змінні
1	2	3
1	$y = \sin \frac{a-x}{c} + 10^4 \sqrt[3]{\frac{a-kx^2}{2b}} + \frac{\cos kx^2}{tg3} - \frac{bc}{ax}$	$a = -1.3; b = 0.91;$ $c = 0.75; x = 2.32; k = 8.$
2	$y = -\frac{(x-d)(x^2+b^2)}{\sqrt[3]{x^2+b^2-cd}} + 10^{-3} tgkn - \frac{\cos kx}{\sin 5}$	$d = 1.25; b = 0.75; n = 4;$ $c = 2.2; x = 0.32; k = 2.$
3	$y = tgik + 10^3 e^{-5} + \sqrt[3]{\frac{10^2 xk }{(a+b)^2}} - \frac{ax^3 - b}{(a+b)^2}$	$i = 5; b = 2.35;$ $a = 25.2; x = 0.1; k = -2.$
4	$y = \frac{\sqrt{ c-d + (a+c)^2}}{\sin 2i} + 10^{-3} e^{ix} - \frac{ c-d + a^2}{\sqrt[3]{(a+c)^2}}$	$a = -1.25; d = 2.5; i = 5;$ $c = 0.05; x = 1.35.$
5	$y = \frac{\ln kx }{\sin 7} - \sqrt{ x-a^2 } - \frac{10^4 a - b}{\cos kx} + \sqrt[3]{x-a^2} + c^3 x$	$a = 0.93; b = 5.61;$ $c = 0.31; x = -2.5; k = 2.$
6	$y = 10^4 \frac{ax}{b^2} - \left \frac{a-b}{kx} \right + \frac{\ln 3}{\sqrt[3]{ax^2 + b^2}} - e^{-kx}$	$b = 0.35;$ $a = 3.5; x = 1.523; k = -2.$

Продовження табл.2.8

1	2	3
7	$y = -\frac{ b-a }{kx} + 10^4 \sqrt[5]{ \cos kx } + \sqrt{\frac{abc}{2.4}} - \frac{0.7abc}{\sin 7}$.	$a = 1.7; b = -1.25;$ $c = -0.3; x = 2.5; k = 3.$
8	$y = \frac{ a^2 - b^2 }{\sin kx} + 10^4 \sqrt[5]{ \sin kx - bc } - \frac{k^2 + tg 3k}{e^{kx}}.$	$a = 1.3; b = 2.42;$ $c = 0.83; x = 1.5; k = 2.$
9	$y = \frac{\sqrt[3]{\ln x + a^2}}{0.47x^2} - \left 0.47x^2 - \frac{10^4}{7} \cos^2 k \right - \frac{c}{x}$	$c = 1.52;$ $a = -2.4; x = 0.29; k = 3.$
10	$y = \frac{1.5(a-b)^2}{ a-b c} + \frac{i}{5} + 10^3 \sqrt{ a-b } - \frac{(a+x^2)\cos 7}{ix^2 + a^2bc}$	$a = -2.5; b = 1.35; i = 3;$ $c = -0.72; x = 2.75.$

Завдання 7. Обчислити за заданою формулою за заданих значень параметрів:

$$ax^2 + bx + c = 0.$$

Параметри, що входять до формули для кожного варіанта наведені в табл. 2.9.

Таблиця 2.9

Варіанти завдань

Варіант	Параметр		
	<i>a</i>	<i>b</i>	<i>c</i>
1	0,56	$1,2 \cdot 10^{-4}$	4,8
3	1,0	0,1	100
5	$4,2 \cdot 10^3$	$8,03 \cdot 10^{-4}$	1,06
7	$7,1 \cdot 10^3$	$9,04 \cdot 10^{-4}$	$8,03 \cdot 10^{10}$
9	5,09	4,32	256

Варіант	Параметр		
	<i>a</i>	<i>b</i>	<i>c</i>
2	4,3	10,7	$3,4 \cdot 10^3$
4	13	0,8	287
6	6,035	5,2	875
8	2,3	7,9	324
10	1,0	0,02	16,57

2.2. Операції з векторами і матрицями в MATLAB

Вивчення способів реалізації засобами системи MATLAB основних операцій із векторами і матрицями.

2.2.1. Введення векторів і матриць

За замовчуванням усі числові змінні в MATLAB вважають матрицями, через це скалярна величина є матрицею першого порядку, а вектори є матрицями, що складаються з одного стовпчика або одного рядка. Матрицю можна ввести, задавши її елементи або прочитавши дані з файлу, а також у результаті звернення до стандартної або написаної користувачем функції [4-7].

Матричні дані розміщені в пам'яті послідовно по стовпцях. Елементи матриці в межах рядка відокремлені пробілами або комами.

Для введення масивів (векторів або матриць) їхні елементи укладають у квадратні дужки []. Так, для введення вектора-рядка розміром 1×3 використовують команду, показану на рис. 2.6.

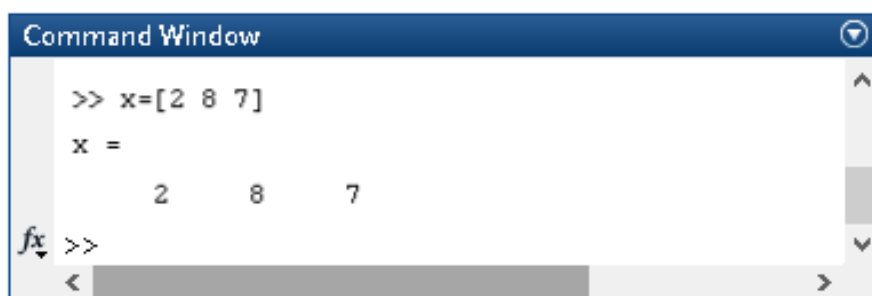


Рис. 2.6. Введення вектора-рядка

Із введенням вектора-стовпця елементи розділяють крапкою з комою (рис. 2.7).

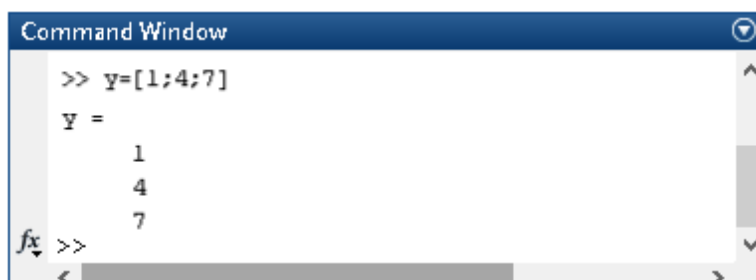


Рис. 2.7. Введення вектора-стовпця

Вводити невеликі за розміром матриці зручно прямо з командного рядка. Із введенням матрицю можна розглядати як вектор-стовпець, кожен елемент якого є вектором-рядком (рис. 2.8),

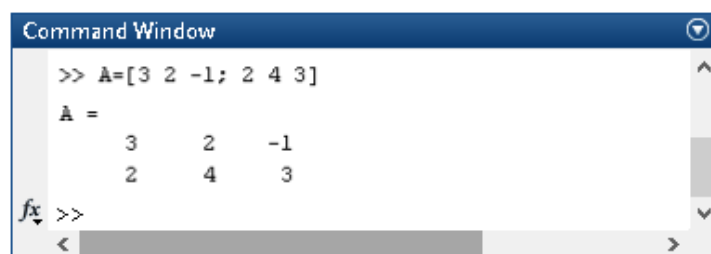


Рис. 2.8. Введення матриці способом 1

або матрицю можна трактувати як вектор-рядок, кожен елемент якої є вектором-стовпцем (рис. 2.9).

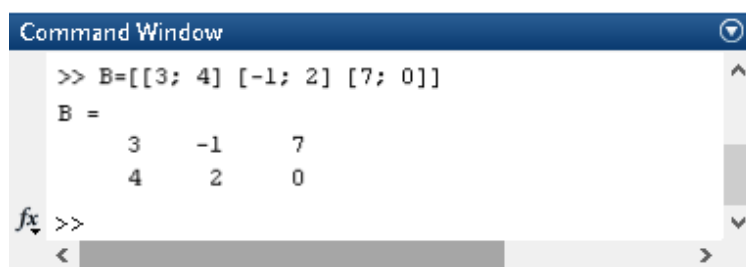


Рис. 2.9. Введення матриці способом 2

Вектори можуть бути сформовані як діапазони — за допомогою двокрапки (:), яка відокремлює початкове значення, крок і кінцеве значення.

Якщо величина кроку відсутня то, за замовчуванням, його значення дорівнює 1.

У результаті команди $a=n:m:k$ буде сформований вектор, останній елемент якого не більше k для додатного m і не менше k – для від’ємного m [n, n+m, n+m+m,...] (рис. 2.10).

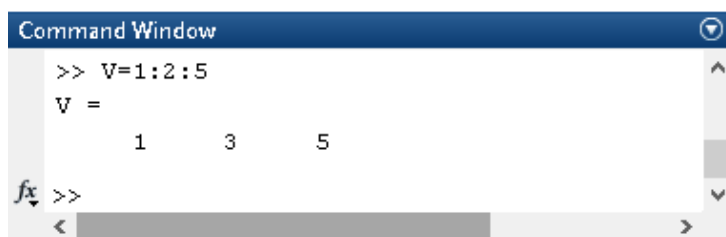


Рис. 2.10. Введення вектора за допомогою оператора діапазону

Задавання діапазону використовують також для організації циклу.

2.2.2. Створення матриць спеціального виду

2.2.2.1. Функції створення матриць спеціального виду. У табл. 2.10 подано деякий набір функцій для створення матриць спеціального виду.

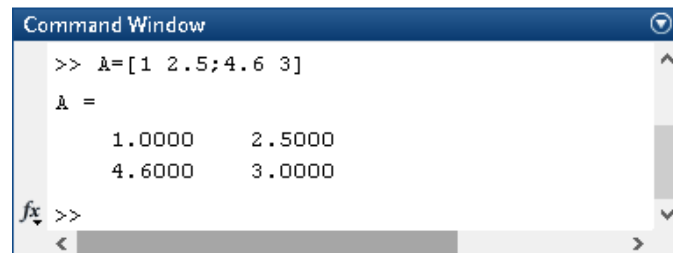
Таблиця 2.10

Функції опису матриць

Функція	Опис
eye(m,n)	Одинична матриця розмірності $m \times n$
zeros(m,n)	Нульова матриця розмірності $m \times n$
ones(m,n)	Матриця, що складається з одних одиниць розмірності $m \times n$
rand(m,n)	Повертає матрицю випадкових чисел, рівномірно розподілених у діапазоні від 0 до 1, розмірність $m \times n$
randn(m, n)	Повертає матрицю розмірності $m \times n$, що складається з випадкових чисел, що мають розподіл за законом Гауса
tril(A), triu(A)	Виділення нижньої трикутної і верхньої трикутної частин матриці A
inv(A)	Знаходження оберненої матриці A
det(A)	Знаходження визначника (детермінанта) квадратної матриці A

Звернення до елемента матриці здійснюється за таким правилом: у круглих дужках після імені матриці вказують індекси, які мають бути додатними цілими числами, що вказують номер рядка, і через кому номер стовпця. Наприклад, $A(2,1)$ означає елемент із другого рядка першого стовпчика матриці A .

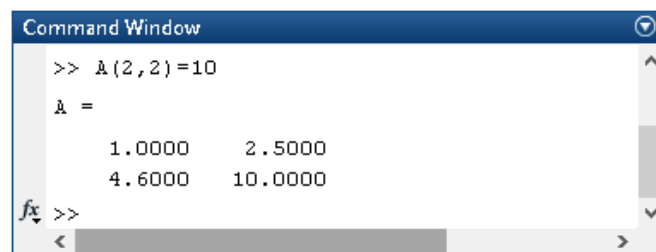
Для подальших прикладів введемо матрицю 2x2 (рис. 2.11).



```
Command Window
>> A=[1 2.5;4.6 3]
A =
    1.0000    2.5000
    4.6000    3.0000
fx >>
```

Рис. 2.11. Введення матриці для прикладів

Щоб змінити елемент матриці, йому потрібно присвоїти нове значення (рис. 2.12).



```
Command Window
>> A(2,2)=10
A =
    1.0000    2.5000
    4.6000   10.0000
fx >>
```

Рис. 2.12. Присвоєння значення елементу матриці

Розмір матриці можна уточнити командою **size**, а результат команди **size** можна використовувати для організації нової матриці.

Приклад 1. Сформувати нульову матрицю того самого порядку, що і матриця A (рис. 2.13).

За допомогою двокрапки легко виділити частину матриці.

Приклад 2. Сформувати вектор із перших двох елементів другого стовпця матриці A (рис. 2.14).

```

Command Window
>> A1=zeros(size(A))
A1 =
     0     0
     0     0
fx >>

```

Рис. 2.13. Вигляд екрана прикладу 1

```

Command Window
>> A2=A(1:2,2)
A2 =
     2.5000
    10.0000
fx >>

```

Рис. 2.14. Вигляд екрана прикладу 2

Двокрапка сама по собі означає рядок або стовпець цілком. Для видалення елемента вектора достатньо присвоїти йому порожній масив - пару квадратних дужок []. Щоб викреслити один або кілька рядків (стовпців) матриці, потрібно вказати діапазон рядків (стовпців, які видаляють) для однієї розмірності і поставити двокрапку для іншої розмірності. Для знаходження довжини вектора можна скористатися також командою **length** [4-7].

Розглянемо приклади використання функцій опису матриць.

1. Виділення нижньої трикутної (рис. 2.15, а) і верхньої трикутної (рис. 2.15, б) частин матриці A.

```

>> A=[1 2 3 4 5 6;7 8 9 10 11 12;13 14 15 16 17 18];

>> tril(A)
ans =
     1     0     0     0     0     0
     7     8     0     0     0     0
    13    14    15     0     0     0

>> triu(A)
ans =
     1     2     3     4     5     6
     0     8     9    10    11    12
     0     0    15    16    17    18

```

а
б

Рис. 2.15. Виділення нижньої трикутної і верхньої трикутної частин матриці A

2. Визначення оберненої матриці, діагоналі матриці та визначника (детермінанта) матриці (рис. 2.16).

```
>> A=[1 2 3 4;5 9 7 8;6 13 11 12;8 5 3 4];
```

Зворотна матриця	Діагональ матриці	Визначник матриці
<pre>>> inv(A) ans =</pre>	<pre>>> diag(A) ans =</pre>	<pre>>> det(A) ans =</pre>
0	1	-48.0000
-0.5000	9	
0.2500	11	
0.2500	4	
-0.3333		
1.1667		
-0.5833		
-0.2500		
-0.3333		
-3.3333		
2.1667		
0.5000		
0.6667		
2.0417		
-1.3958		
-0.3125		

Рис. 2.16. Визначення оберненої матриці, діагоналі матриці та визначника матриці

2.2.2.2. Арифметичні операції для роботи з матрицями. Набір арифметичних операцій у MATLAB для роботи з матрицями складається зі стандартних операцій: додавання, віднімання, множення, ділення, піднесення до степеня, і доповнений спеціальними матричними операціями (табл. 2.11). Якщо операція застосована до матриць, розміри яких не узгоджені, то буде виведено повідомлення про помилку [4-7].

Таблиця 2.11

Знаки операцій

Символ	Призначення
1	2
+, -	Символи плюс і мінус означають знак числа або операцію додавання і віднімання матриць, причому матриці мають бути однієї розмірності
*	Знак множення означає матричне множення, для поелементного множення матриці застосовують комбінований знак (.*)

1	2
'	Апостроф означає операцію транспонування (разом із комплексним сполученням), транспонування без обчислення сполучення позначають за допомогою комбінованого знака (.')
/	Ділення «зліва направо»
\	Ділення «справа наліво»
^	Оператор піднесення до степеня, для поелементного піднесення до степеня застосовують комбінований знак (.^)

Для поелементного виконання операцій множення, ділення і піднесення до степеня застосовують комбіновані знаки (**крапка і знак операції**). Наприклад, якщо за матрицею стоїть знак (^), то вона піднесена до степеня, а комбінація (.^) означає піднесення до степеня кожного елемента матриці. Із множенням (додаванням, відніманням, діленням) матриці на число відповідна операція завжди проводиться **поелементно**.

Приклад 3. Звичайне і поелементне множення матриць розміром 2x2 (рис. 2.17).

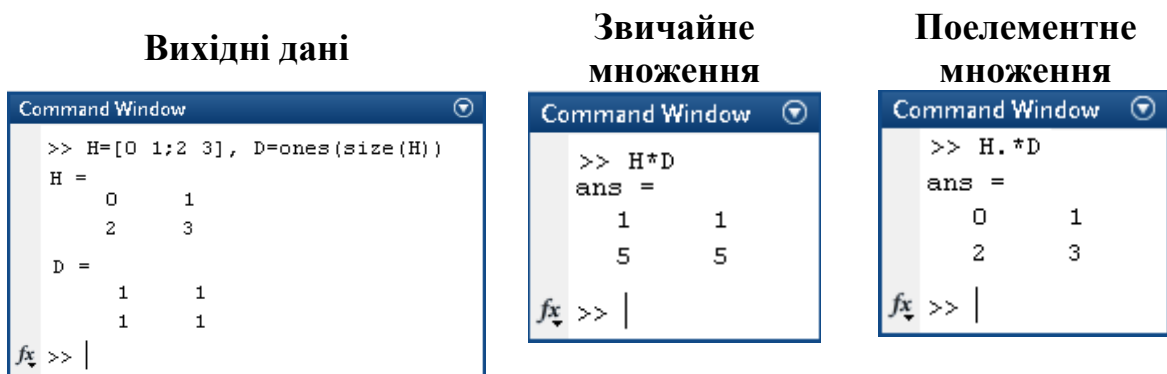


Рис. 2.17. Результати виконання прикладу 3

2.2.3. Функції для розв'язання задач матричної алгебри

Система MATLAB має ряд функцій, призначених для обробки даних, заданих у матричній або векторній формі (розв'язання задач матричної алгебри) (табл. 2.12) [4-7].

Функції MATLAB для розв'язання задач матричної алгебри

Функція	Опис
1	2
$A = [a_{11} \ a_{12} \ \dots \ a_{1n}; \dots; a_{m1} \ a_{m2} \ \dots \ a_{mn}]$	Визначення матриці розміру $m \times n$. Рядки відокремлюють крапкою з комою, а елементи кожного рядка – знаком пробілу. <i>Наприклад,</i> <pre>>> A=[1 -3 2; 4 3 8; -2 6 1]</pre> <pre>A =</pre> <pre> 1 -3 2</pre> <pre> 4 3 8</pre> <pre> -2 6 1</pre> <pre>>> B=[-1;2;6]</pre> <pre>B =</pre> <pre> -1</pre> <pre> 2</pre> <pre> 6</pre>
A' (апостроф)	Транспонування матриці A. <i>Наприклад,</i> B – стовпець, у B - рядок: <pre>>> B=B'</pre> <pre>B =</pre> <pre> -1 2 6</pre>
A1+A2	Сума двох матриць (матриці повинні мати однаковий розмір). <i>Наприклад,</i> <pre>>> A1=[3 4;1 -2;0 -7] >> A2=[2 -5;-9 -6; 8 5] >> A1+A2</pre> <pre>A1 =</pre> <pre> 3 4</pre> <pre> 1 -2</pre> <pre> 0 -7</pre> <pre>A2 =</pre> <pre> 2 -5</pre> <pre> -9 -6</pre> <pre> 8 5</pre> <pre>ans =</pre> <pre> 5 -1</pre> <pre> -8 -6</pre> <pre> 18 -2</pre>
$a*A1 \pm b*A2$	Операції з комбінацією матриць (матриці повинні мати однаковий розмір), a і b – це числа. <i>Наприклад,</i> <pre>>> 3*A1-2*A2</pre> <pre>ans =</pre> <pre> 5 22</pre> <pre> 21 6</pre> <pre> -16 -31</pre>
f(A)	Якщо f(x) – це елементарна функція, то визначена операція призводить до обчислення матриці того самого розміру, що і матриця A, кожний елемент якої дорівнює значенню елементарної функції від відповідного елемента матриці A. <i>Наприклад,</i> <pre>>> sin(A1) >> exp(B)</pre> <pre>ans =</pre> <pre> 0.1411 -0.7568 0.3679 7.3891 403.4288</pre> <pre> 0.8415 -0.9093</pre> <pre> 0 -0.6570</pre>

1	2
size(A)	Повертає масив, що складається з кількості рядків і числа стовпців матриці: <pre>>> C=[2 1 0;3 1 3;4 -2 -7]; >> C C = 2 1 0 3 1 3 4 -2 -7</pre> <pre>>> size(C) ans = 3 3</pre>
sum(A)	Повертає суму всіх елементів по стовпцю (вектору)
mean(A)	Повертає середнє значення стовпця матриці (вектора)
std(A)	Повертає середньоквадратичне відхилення стовпця матриці (вектора)
min(A), max(A)	Повертає мінімум і максимум відповідно по стовпцю матриці (вектору)
sort(A)	Сортує стовпець матриці за зростанням (вектора)
prod(A)	Обчислює добуток усіх елементів стовпців (вектора)
diag(B)	Створення квадратної матриці, на діагоналі якої стоять елементи вектора B, а інші елементи дорівнюють 0. Наприклад, <pre>>> diag(B) ans = -1 0 0 0 2 0 0 0 6</pre>
prod(x)	Обчислення добутку елементів вектора. Наприклад, <pre>>> prod(B) ans = -12</pre>
A(i,j)	Звернення до елемента матриці, який розташований в <i>i</i> -му рядку та <i>j</i> -му стовпці. Наприклад, <pre>>> C(2,3) >> B(2) ans = ans = 3 2</pre>
A(i,:),A(:,j)	Звернення до елементів матриці, які розташовані в <i>i</i> -му рядку або <i>j</i> -му стовпці. Наприклад, <pre>>> C(:,1) ans = 2 3 4</pre> <pre>>> C(2,:) ans = 3 1 3</pre>

Символи і текстові рядки в MATLAB вводять за допомогою простих лапок (' '). У внутрішньому поданні символи задані цілими числами. Конвертувати масив символів у числову матрицю дозволяє команда **double**. Зворотна операція відбувається за командою **char**. Друковані символи зі стандартного набору ASCII подані числами від 32 до 255.

Для подання числових даних у рядкові змінні є ряд команд перетворення. У табл. 2.13 наведені деякі функції для цих і зворотних операцій, а повний список команд можна отримати за командою **help strfun** [4-7].

Таблиця 2.13

Функції роботи із рядковими змінними

Функція	Дія
num2str	Подання числа в рядок
int2str	Подання цілого числа в рядок
mat2str	Перетворення матриці в рядок
str2mat	Об'єднання рядків у матрицю
str2num	Перетворення рядка в число
strcat	Об'єднання рядків

ЗАВДАННЯ ДЛЯ ВИКОНАННЯ

Завдання 1. Введення з клавіатури векторів і матриць. Ввести:

- довільний вектор-рядок (**v**) розмірністю 3;
- довільний вектор-стовпець (**w**) розмірністю 3;
- довільну матрицю (**m**) розмірністю 3×3.

Завдання 2. Генерація матриць спеціального виду. Створити:

- матрицю з нульовими елементами (**m0**) розмірністю 4×4;
- матрицю з одиничними елементами (**m1**) розмірністю 4×4;
- матрицю з елементами, що мають випадкові значення (**mr**) розмірністю 4×4;

- матрицю з одиничними діагональними елементами (***me***) розмірністю 4×4 .

Завдання 3. Обчислити матрицю ***M*** за формулою свого варіанта (табл. 2.14). Якщо матриці і вектори різного розміру, зробити заміну в завданні 1.

Завдання 4. Провести такі обчислення, використовуючи функції обробки даних:

- визначити кількість рядків і стовпців матриці ***M***;
- визначити максимальний елемент матриці ***M***;
- визначити мінімальний елемент матриці ***M***;
- підсумувати елементи матриці ***M***;
- перемножити елементи матриці ***M***;
- поміняти місцями перший і другий стовпчики матриці ***M***;
- обчислити визначник отриманої матриці та порівняти його з визначником вихідної матриці (***M***).

Завдання 5. Самостійно задати матриці та перевірити рівність $(AB)^T B^T = B^T A^T$ для матриць розмірами 3×3 і 5×5 ; перевірити, чи існує обернена матриця до матриці ***A*** (визначник не дорівнює 0), і, якщо існує, знайти її та вивести результат у командне вікно так, щоб елементи оберненої матриці були подані як звичайні дробки.

Таблиця 2.14

Варіанти завдання 3

Варіант	Завдання	Варіант	Завдання
1	$M = v * w + m + mr * me$	6	$M = m * w + mr * v'$
2	$M = m + mr * me$	7	$M = m * mr + w * v$
3	$M = (v/m) * (mr + me)$	8	$M = m + mr - 100$
4	$M = w * v + mr * me$	9	$M = v' + w + mr * w$
5	$M = m * mr + me$	10	$M = m + m1 * me$

Завдання 6. Виконати операції з матрицями за власним варіантом із табл. 2.15.

Таблиця 2.15

Варіанти завдання 6

Варіант	Завдання	Варіант	Завдання
1	2	3	4
1	Видалити з матриці A третій стовпець, а з матриці B другий рядок. У матриці A змінити значення елемента з координатами (2, 1) на 6. Розрахувати суму рядків, стовпців і діагоналі матриці B. $A = \begin{bmatrix} 7 & 2 & 5 \\ 10 & 4 & 6 \\ 3 & 5 & 9 \end{bmatrix} \quad B = \begin{bmatrix} 5 & 9 & 0 \\ 3 & 12 & 4 \\ 1 & 4 & 7 \end{bmatrix}$	2	Видалити перший рядок і третій стовпець «магічної» матриці M розміром 4 x 4. Отриману матрицю скласти з матрицею A, у якій заздалегідь необхідно обернути рядки на стовпці, а стовпці на рядки. $A = \begin{bmatrix} 6 & 7 & 9 \\ 4 & 8 & 3 \\ 1 & 0 & 5 \end{bmatrix}$
3	Задати «магічну» матрицю A розміром 4 x 4. Створити матрицю B, кожний елемент якої є синусом відповідного елемента «магічної» матриці A. Розрахувати суму рядків, стовпців і діагоналі отриманої матриці B	4	Обчислити результат добутку елемента матриці M із координатами (2, 3) на синус значень першого стовпця цієї ж матриці. $M = \begin{bmatrix} 1 & 2 & 4 & 6 \\ 3 & 7 & 4 & 9 \\ 7 & 8 & 11 & 0 \\ 8 & 5 & 6 & 4 \end{bmatrix}$
5	Створити матрицю A розміром 3 x 6 із випадковими елементами, що розподілені за законом Гауса. Повернути цю матрицю проти годинникової стрілки на 90°	6	Створити матрицю A розміром 2 x 4 з одиницями по головній діагоналі та всіма іншими елементами – нулями. Змінити розмір матриці на 8 x 1

1	2	3	4
7	Створити матрицю A з нульовими елементами розміром 5 x 5 та одиничну матрицю B розміром 2 x 3. Замінити верхню центральну частину матриці A елементами матриці B	8	Створити матрицю A з випадковими елементами розміром 4 x 5. Переставити стовпці відносно вертикальної осі, а рядки переставити відносно горизонтальної осі
9	На основі матриці M утворити нижню трикутну матрицю і повернути її проти годинникової стрілки на 90°. $M = \begin{bmatrix} 1 & 3 & 4 & 6 \\ 3 & 4 & 1 & 2 \\ 9 & 7 & 6 & 5 \end{bmatrix}$	10	Створити матрицю A , у якій головною діагоналлю є вектор $V = [2 \ 3 \ 6 \ -4]$. Обернути отриману матрицю

2.3. Використання основних операторів у математичних розрахунках і робота з масивами (матрицями)

Обчислення різних математичних виразів, виконання різних дій над масивами.

Масиви – основні об'єкти в системі MATLAB: **одновимірні масиви – вектори**, а **двовимірні - матриці**; у версіях вище 5.0 можливе використання **багатовимірних масивів – тензорів** [4-7].

Порядок роботи з невеликими матрицями (введення, операції над матрицями і векторами тощо) розглянуто в попередньому пункті.

Розглянемо інші види матриць.

2.3.1. Блочні матриці

Часто в додатках виникають так звані блочні матриці, тобто матриці, складені з підматриць (блоків). Розглянемо спочатку їх.

Приклад 1. Створити блочну матрицю $K = \begin{pmatrix} A & B \\ C & D \end{pmatrix}$ із відповідних матриць розміром 2x2.

Ураховуючи, що матриця складається з двох рядків, у першому рядку матриці A і B , а в другому – C і D , блочну матрицю можна сформувати так:

$$K = [A \ B; \ C \ D].$$

Якщо вважати, що матриця складається з двох стовпців, у першому - матриці A і C , а в другому – B і D :

$$K = [[A;C],[B;D]].$$

Результат показано на рис. 2.18.

```

Command Window
>> A=[-1,4;-1,4]
A =
    -1     4
    -1     4
>> B=[2,0;0,5]
B =
     2     0
     0     5
>> C=[3,-3;-3,3]
C =
     3    -3
    -3     3
>> D=[8,9; 1,10]
D =
     8     9
     1    10
fx >>

Command Window
>> K=[ [A;C] , [B;D] ]
K =
    -1     4     2     0
    -1     4     0     5
     3    -3     8     9
    -3     3     1    10
fx >>

```

Рис. 2.18. Створення блочної матриці

Зворотним завданням для конструювання блочних матриць є завдання виділення блоків. Виділяють блоки матриць індексацією за допомогою двокрапки:

1. Для виділення з матриці підматриці задають номери тих рядків і стовпців, які потрібно виділити, за допомогою двокрапки (рис. 2.19).

$$P1 = P(2:3,2:3);$$

2. Для виділення з матриці стовпчика або рядка (тобто масиву, у якого один із розмірів дорівнює одиниці) потрібно як один з індексів використовувати номер стовпчика або рядка матриці, а інший індекс замінити двокрапкою без зазначення інтервалу:

$$p = P(2,:).$$

3. Виділяючи блок, до кінця матриці можна не вказувати її розміри, а використовувати елемент **end** (рис. 2.19).

$$p = P(2, 2:end).$$

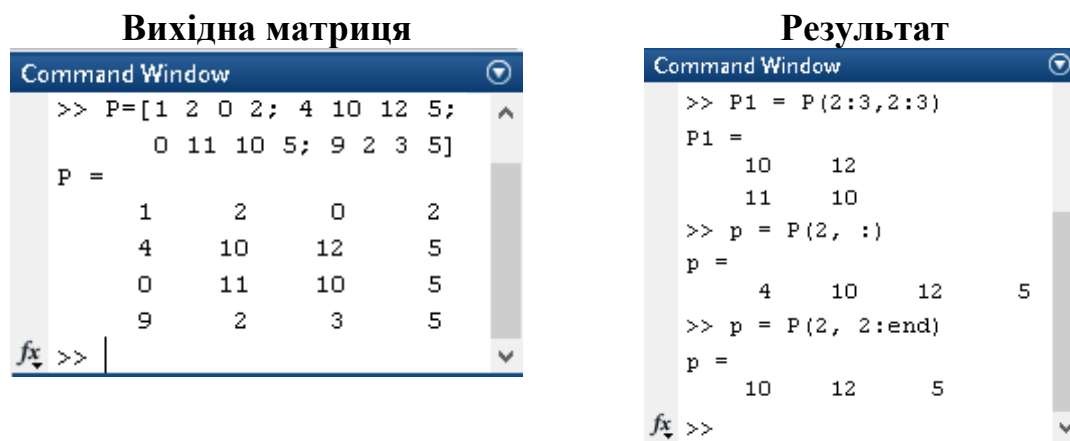


Рис. 2.19. Виділення із матриці підматриць

2.3.2. Видалення рядків і стовпців

У MATLAB парні квадратні дужки [] означають порожній масив, який, зокрема, дає змогу видаляти рядки і стовпці матриці. Для видалення рядка потрібно присвоїти йому порожній масив [4-7].

Приклад 2. Із матриці **P** видалити:

- 1) перший рядок - $P(1,:)=[];$
- 2) визначити розмір утвореного масиву - $\text{size}(P);$
- 3) видалити другий і третій стовпчики $P(:, 2:3) = [].$

Результат наведений на рис. 2.20.

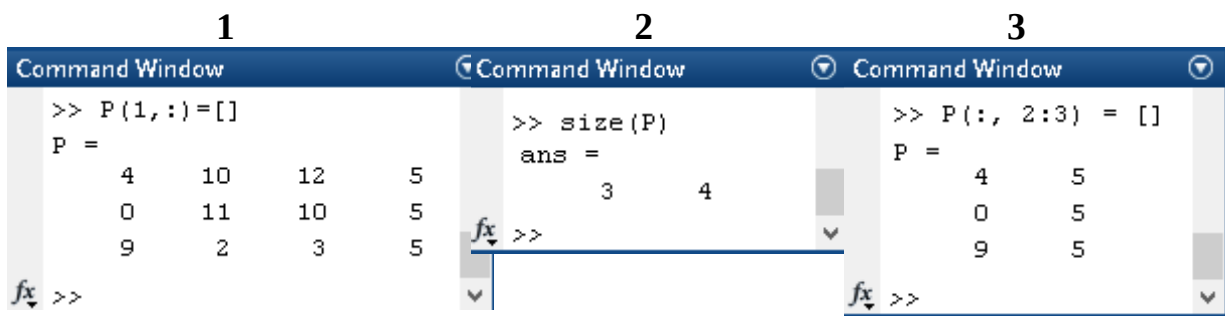


Рис. 2.20. Видалення із матриці рядків і стовпців (приклад 2)

2.3.3. Заповнення матриць за допомогою індексації

Індексація суттєво економить час для введення матриць, що мають певну структуру.

Вище було описано кілька способів введення матриць у MATLAB. Однак часто буває простіше згенерувати матрицю, ніж вводити її, особливо якщо вона має просту структуру. Розглянемо приклад такої матриці.

Приклад 3. Створити матрицю за допомогою індексації:

$$A = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & -1 & -1 \end{bmatrix}$$

Генерація матриці T здійснюється в три етапи:

1. Створення масиву T розміром 5×5 , що складається з нулів.
2. Заповнення першого рядка одиницями.
3. Заповнення частини останнього рядка мінус одиницями до останнього елемента.

Результати наведені на рис. 2.21.

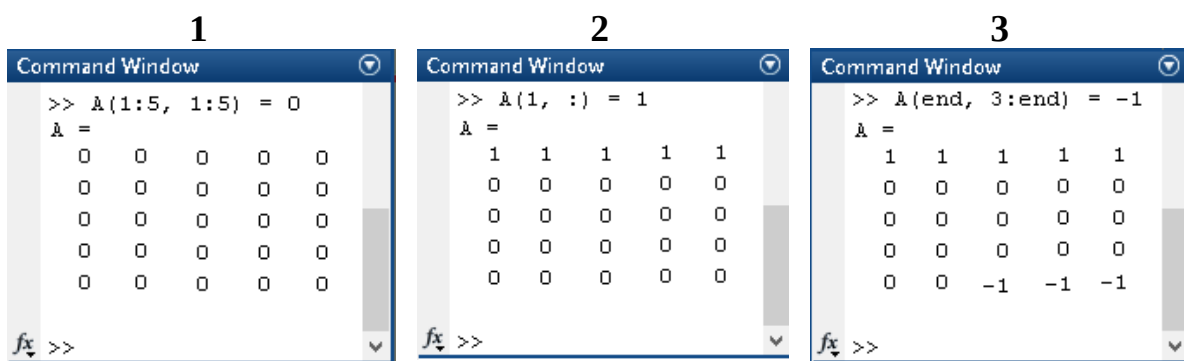


Рис. 2.21. Видалення з матриці рядків і стовпців (приклад 3)

2.3.4. Створення матриць спеціального виду

Формують матриці спеціального виду за допомогою **спеціальних функцій** (табл. 2.16).

Таблиця 2.16

Функції для створення матриць спеціального виду

Функція	Значення
inv	Зворотна матриця
det	Знаходження визначника (детермінанта) квадратної матриці
zeros	Формування масиву нулів
ones	Формування масиву одиниць
eye	Формування одиничної матриці
rand	Формування масиву елементів, розподілених за рівномірним законом
randn	Формування масиву елементів, розподілених за нормальним законом
cross	Векторний добуток
kron	Формування тензорного добутку
linspace	Формування лінійного масиву рівновіддалених вузлів
logspace	Формування вузлів логарифмічної сітки
meshgrid	Формування вузлів двовимірної і тривимірної сіток

Над створеними матрицями можна проводити **операції** (табл. 2.17) [4-7].

Таблиця 2.17

Операції над матрицями

Операція	Дія, яку виконують
diag	Формування або виділення діагоналей матриці
tril	Формування нижньотрикутної матриці (масиву)
triu	Формування верхньотрикутної матриці (масиву)
fliplr	Поворот матриці відносно вертикальної осі
flipud	Поворот матриці відносно горизонтальної осі
rot90	Поворот матриці на 90 градусів
reshape	Перетворення розмірів матриці

Крім того, існують **спеціальні матриці** (табл. 2.18).

Таблиця 2.18

Спеціальні матриці

Ім'я	Матриця
companion	Супроводжувальна матриця характеристичного многочлена
hadamard	Матриця Адамара (Hadamard matrix)
hankel	Матриця Ганкеля (Hankel matrix)
hilb, invhild	Матриця Гільберта (Hilbert matrix)
magic	Магічний квадрат (сума чисел у рядках, стовпцях, діагоналях - однакова)
pascal	Матриця Паскаля (Pascal matrix)
rosser	Матриця Рессера (Rosser matrix)
toeplitz	Матриця Теплиця (Toeplitz matrix)
vander	Матриця Вандермонда (Vandermonde matrix)
wilkinson	Матриця Уілкінсона (Wilkinson matrix)

Більш детальну інформацію про функції, операції, спеціальні матриці (структура, формат, приклад використання) можна отримати за допомогою команди

help <ім'я>.

Примітка. Використання одного аргументу у функціях призводить до створення відповідної квадратної матриці.

Приклад 4. Створити матриці, наведені на рис. 2.22.

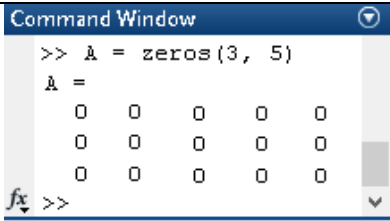
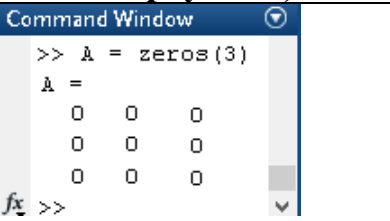
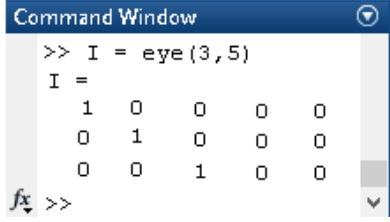
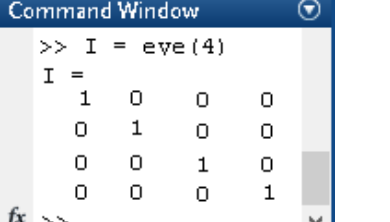
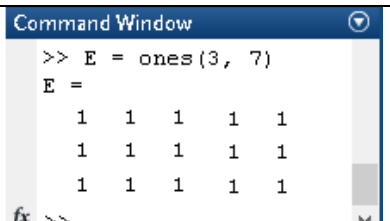
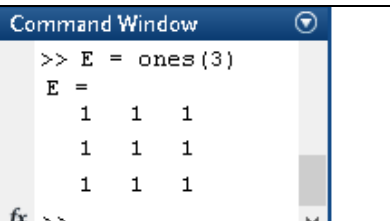
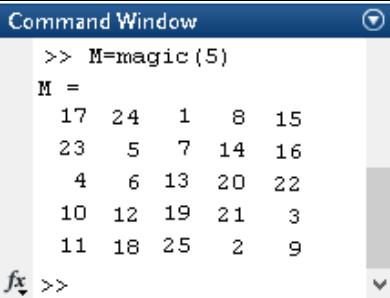
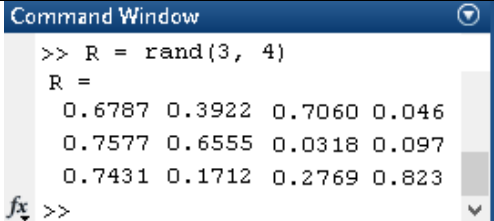
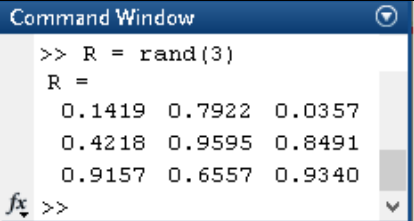
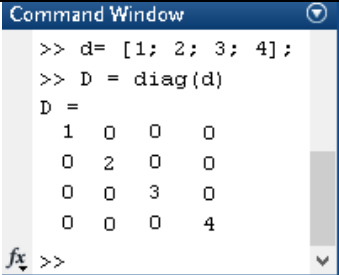
Завдання	Результат (два аргументи)	Результат (один аргумент)
Заповнити матрицю нулями		
Створити одиничну матрицю		
Створити матрицю із одиниць		
Створити магічний квадрат		
Заповнити матрицю випадковими числами		
Створити діагональну матрицю		

Рис. 2.22. Приклади створення матриць

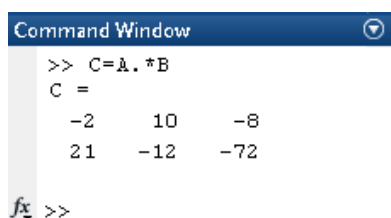
2.3.5. Поелементні операції з матрицями

Оскільки вектори та матриці зберігаються у двовимірних масивах, то застосування математичних функцій до матриць і поелементні операції проводять так само, як для векторів [4-7].

Для демонстрації поелементних операцій з матрицями введемо дві матриці

$$A = \begin{pmatrix} 2 & 5 & -1 \\ 3 & 4 & 9 \end{pmatrix} \quad \text{і} \quad B = \begin{pmatrix} -1 & 2 & 8 \\ 7 & -3 & -5 \end{pmatrix}.$$

2.3.5.1. Множення кожного елемента однієї матриці на відповідний елемент інший відбувається за допомогою оператора `.*` (рис. 2.23).

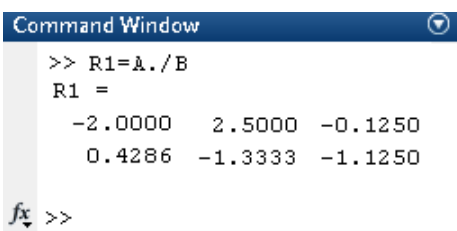


```
Command Window
>> C=A.*B
C =
    -2     10     -8
    21    -12    -72
fx >>
```

Рис. 2.23. Поелементне множення матриць

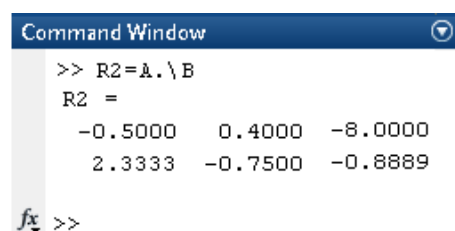
2.3.5.2. Для ділення елементів першої матриці на відповідні елементи другої використовують оператор `./` (рис. 2.24, а), а для ділення елементів другої матриці на відповідні елементи першої - `.\` (рис. 2.24, б). Для звичайного ділення матриці на матрицю служить `/` (рис. 2.24, в).

а



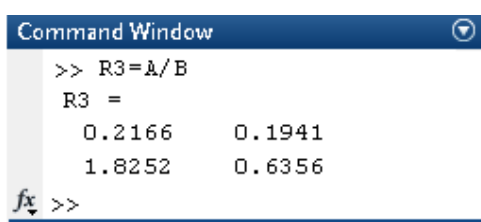
```
Command Window
>> R1=A./B
R1 =
   -2.0000    2.5000   -0.1250
    0.4286   -1.3333   -1.1250
fx >>
```

б



```
Command Window
>> R2=A.\B
R2 =
   -0.5000    0.4000   -8.0000
    2.3333   -0.7500   -0.8889
fx >>
```

в



```
Command Window
>> R3=A/B
R3 =
    0.2166    0.1941
    1.8252    0.6356
fx >>
```

Рис. 2.24. Ділення матриць

2.3.5.3. Поелементне піднесення до степеня здійснюють за допомогою оператора $\wedge$. Показник степеня може бути числом чи матрицею того самого розміру, що і матриця, яку підносять до степеня [4-7].

ЗАВДАННЯ ДЛЯ ВИКОНАННЯ

Завдання 1. Обчислити значення функції $f(x)$ на інтервалі $[a; b]$ із кроком h . У табл. 2.19 наведені варіанти завдань.

Таблиця 2.19

Варіанти завдання 1

Варіант	Функція $f(x)$	Параметр		
		a	b	h
1	$\frac{x^2}{1+0,25\sqrt{x}}$	1,1	3,1	0,2
3	$\frac{\cos \pi x^2}{\sqrt{1-3x}}$	-1	0	0,1
5	$\frac{e^{x/3}}{1+x^2}$	1,4	2,4	0,1
7	$\sqrt{3+2x} \cdot \operatorname{tg} \frac{\pi x^3}{2}$	0,1	0,9	0,08
9	$\sqrt[3]{x^2+3} \cdot \cos \frac{\pi x}{2}$	1	1,5	0,15
2	$\frac{x^3-0,3x}{\sqrt{1+2x}}$	2,05	3,05	0,1
4	$\frac{2e^{-x}}{2\pi+x^3}$	0	1,6	0,16
6	$\sqrt{1+4x} \cdot \sin \pi x$	0,1	0,8	0,05
8	$e^{-2x} + x^2 - 1$	0,25	2,25	0,25
10	$(e+x) \cdot \sin(\pi\sqrt{x-1})$	1,8	2,8	0,1

Завдання 2. Обчислити значення функції відповідно до свого варіанта (табл. 2.20).

Завдання 3. Зробити заміну змінних на відповідний вираз (табл. 2.20). Скласти нову функцію, присвоїти їй нове ім'я і обчислити її значення.

Завдання 4. Створити квадратну матрицю M розміром 5×5 . Елементи матриці – додатні і від'ємні двозначні десяткові числа. Обчислити і вивести на екран її визначник, максимальне і мінімальне значення.

Варіанти завдань 2-4

Варі- ант	Вираз / Значення	Заміна
1	$F = \left(\frac{1 + \sqrt{x}}{\sqrt{1+x}} - \frac{\sqrt{1+x}}{1 + \sqrt{x}} \right)^2 - \left(\frac{1 - \sqrt{x}}{\sqrt{1+x}} - \frac{\sqrt{1+x}}{1 - \sqrt{x}} \right)^2$	$\sqrt{1+x} \rightarrow \cos(x)$ $x=5$
	$x=3$	
2	$F = \frac{1,5(a-b)^2}{ a-b c} + \frac{i}{5} + 10^3 \sqrt{ a-b } - \frac{(a+x^2) \cos 7}{ix^2 + a^2 bc}$	$a-b$ $\rightarrow \sqrt{a^2 + b^2}$
	$a=2,5; b=1,35; i=3; c=-0,72; x=2,75$	
3	$O = \frac{\cos^2 x^3}{x^2 + x} + \frac{\sqrt{x^2 + 1}}{ x + 1} + \lg(abc) - \sqrt[4]{\frac{\sqrt{x^2 + 1}}{1 + \sqrt{x^2 + 1}}}$	$\sqrt{x^2 + 1} \rightarrow a/b/c$
	$a=-3,1; b=1,82; c=-4,5; x=1,2$	
4	$F = \left(\frac{1 + \sqrt{x}}{\sqrt{1+x}} - \frac{\sqrt{1+x}}{1 + \sqrt{x}} \right)^2 - \left(\frac{1 - \sqrt{x}}{\sqrt{1+x}} - \frac{\sqrt{1+x}}{1 - \sqrt{x}} \right)^2$	$\sqrt{1+x} \rightarrow \cos(x)$ $x = 2$
	$x=2$	
5	$y = -\frac{(x-d)(x^2 + b^2)}{\sqrt[3]{x^2 + b^2 - cd}} + 10^{-3} \operatorname{tg} kn - \frac{\cos(kx)}{\sin(5)}$	$b^2 \rightarrow d \cos x$ $\cos kx \rightarrow \sin kx$
	$k=-1,3; b=0,91; c=0,75; x=2,32; d=1; n=5$	
6	$y = \frac{ a^2 - b^2 }{\sin kx} + 10^4 \sqrt[5]{ \sin(kx - bc) } - \frac{k^2 + \operatorname{tg} 3k}{e^{kx}}$	$\sin kx \rightarrow \sqrt[3]{x^4}$ $kx \rightarrow a^2 - b^2 $
	$a=1,7; b=-1,25; k=3; x=2,5; c=-0,3$	
7	$y = 10^4 \sin^2 i - \frac{0,32x^3 + 4x + b}{\cos ia} \sqrt[6]{0,32x^3 - b + b }$	$0,32x^3 \rightarrow \sin^2 x$ $+ 1$
	$i=3; x=2,75; a=-2,5; b=1,35$	
8	$y = \frac{1}{9} - 10^{-4} e^{kx}$	$x^2 + b \rightarrow 0,4x$
	$+ \cos \sqrt{x^2 + b} + \frac{\sqrt{x^2 + b}}{0,4x} + \frac{\sin 3}{(x^2 + b)n}$ $k=3; x=0,48; b=-0,31; n=1,72$	
9	$y = \cos(k(x-a)) + 10^{-4} \frac{(x+a)^3 + x^4 d}{k(x-a)^3} + \frac{\sqrt[5]{ x+a }}{2,4b}$	$x-a \rightarrow x+a$ $x+a \rightarrow x-a$
	$k=4; x=8,2; a=-3,25; b=0,05; d=0,95$	
10	$h = \frac{\sqrt[3]{\ln x + a^2}}{0,47x^2} - \left 0,47x^2 - \frac{10^4}{7} \cos^2 k \right - \frac{c}{x}$	$0,47x^2 \rightarrow \sin x^2$
	$a=1,3; x=1; k=2; c=-0,83$	

Завдання 5. Транспонувати матрицю. Перевірити обидві матриці на «магічність». Результати перевірки вивести на екран.

Завдання 6. Створити дві квадратні матриці M_1 і M_2 розмірністю 3×3 . Заповнити матрицю M_1 випадковими значеннями, а M_2 – магічний квадрат. Провести поелементне множення матриць. В одержаній матриці обчислити визначник. Вивести на екран другий стовпець і поелементно третій рядок.

Завдання 7. Створити матрицю розміром 5×5 , елементами якої є випадкові числа. Визначити детермінант матриці, обернену матрицю, сформувати нижньотрикутну та верхньотрикутну матриці.

Завдання 8. За допомогою вбудованих функцій для заповнення стандартних матриць, індексації двокрапкою і, можливо, об'єднання, повернення чи транспонування отримати матриці з табл. 2.21. Застосувати функції обробки даних і поелементні операції для знаходження відповідних величин.

Таблиця 2.21

Варіанти завдання 8

Варіант	Створити матрицю	Обчислити
1	2	3
1	$A = \begin{bmatrix} 1 & 0 & 1 & -3 & -3 \\ 0 & 1 & 1 & -3 & -3 \\ 1 & 0 & 1 & -3 & -3 \\ 0 & 0 & 0 & 4 & 4 \\ 0 & 0 & 0 & 4 & 4 \end{bmatrix}$	$s = \sum_{k=1}^5 a_{kk}^3$
2	$A = \begin{bmatrix} -1 & -1 & 1 & 1 & 1 \\ -1 & -1 & 1 & 1 & 1 \\ -1 & -1 & 1 & 1 & 1 \\ 2 & 2 & 3 & 0 & 0 \\ 2 & 2 & 0 & 0 & 3 \end{bmatrix}$	$s = \sum_{i=1}^5 \min_{j=1, \dots, 5} (a_{ij})$
3	$A = \begin{bmatrix} 1 & 0 & 0 & 4 & 0 & 0 \\ 0 & 1 & 0 & 0 & 4 & 0 \\ 0 & 0 & 1 & 0 & 0 & 4 \\ 2 & 2 & 2 & 3 & 3 & 3 \\ 2 & 2 & 2 & 3 & 3 & 3 \\ 2 & 2 & 2 & 3 & 3 & 3 \end{bmatrix}$	$m = \max \left\{ \sum_{i=1}^6 a_{ij}^2 \right\}$

1	2	3
4	$A = \begin{bmatrix} 2 & 2 & -1 & -1 & 2 & 2 \\ 2 & 2 & -1 & -1 & 2 & 2 \\ 2 & 2 & -1 & -1 & 2 & 2 \\ 2 & 2 & -1 & -1 & 2 & 2 \\ 2 & 2 & -1 & -1 & 2 & 2 \\ 2 & 2 & -1 & -1 & 2 & 2 \end{bmatrix}$	$s = \sum_{i=1}^6 \sum_{j=1}^6 a_{ij} $
5	$A = \begin{bmatrix} 1 & 2 & 3 & 4 & 5 & 6 \\ -1 & -2 & -3 & -4 & -5 & -6 \\ 4 & 4 & 4 & 4 & 4 & 4 \\ 4 & 4 & 4 & 4 & 4 & 4 \\ 5 & 5 & 5 & 5 & 5 & 5 \\ 5 & 5 & 5 & 5 & 5 & 5 \end{bmatrix}$	$m = \min_{i,j=1,\dots,6} a_{ij}^3$
6	$A = \begin{bmatrix} -1 & -1 & -1 & 1 & 1 & 1 \\ -1 & -1 & -1 & 1 & 1 & 1 \\ -1 & -1 & -1 & 1 & 1 & 1 \\ 2 & 2 & 2 & 3 & 0 & 0 \\ 2 & 2 & 2 & 0 & 3 & 0 \\ 2 & 2 & 2 & 0 & 0 & 3 \end{bmatrix}$	$s = \sum_{k=1}^6 a_{kk}^3$
7	$A = \begin{bmatrix} 1 & 1 & -3 & -3 & -3 & -3 \\ 1 & 1 & -3 & -3 & -3 & -3 \\ -3 & -3 & 2 & 0 & 0 & 0 \\ -3 & -3 & 0 & 2 & 0 & 0 \\ -3 & -3 & 0 & 0 & 2 & 0 \\ -3 & -3 & 0 & 0 & 0 & 2 \end{bmatrix}$	$s = \sum_{i=1}^6 a_{1i} + \sum_{i=1}^6 a_{4i}$
8		$s = \sum_{i=1}^6 a_{ii} + \sum_{i=1}^6 a_{i6}$
9	$A = \begin{bmatrix} 0 & 0 & 0 & 0 & 3 & 4 \\ 0 & 0 & 0 & 3 & 0 & 4 \\ 0 & 0 & 3 & 0 & 0 & 4 \\ 0 & 3 & 0 & 0 & 0 & 4 \\ 3 & 0 & 0 & 0 & 0 & 4 \\ 1 & 1 & 1 & 1 & 1 & -2 \end{bmatrix}$	$s = \sum_{i=1}^6 \sum_{j=1}^6 \sin\left(\frac{\pi}{6} a_{ij}^2\right)$
10	$A = \begin{bmatrix} 1 & 2 & 3 & -1 & 0 & 0 \\ 1 & 2 & 3 & 0 & -1 & 0 \\ 1 & 2 & 3 & 0 & 0 & -1 \\ 0 & 0 & 7 & 2 & 2 & 2 \\ 0 & 7 & 0 & 2 & 2 & 2 \\ 7 & 0 & 0 & 2 & 2 & 2 \end{bmatrix}$	$m = \max_{i=1,\dots,6} \min_{j=1,\dots,6} a_{ij}$

2.4. Побудова графіків функцій у MATLAB

Вивчення основних операторів побудови графіків, діаграм, поверхней у системі MATLAB і створення програм, що реалізують графічне виведення інформації для проведення обчислень.

2.4.1. Побудова графіків

Одна з переваг системи MATLAB - велика кількість засобів графіки, починаючи від команд побудови простих графіків функцій однієї змінної в декартовій системі координат і закінчуючи комбінованими і презентаційними графіками з елементами анімації, а також засобами проєктування графічного призначеного для користувача інтерфейсу (GUI). Особливу увагу в системі приділено тривимірній графіці з функціональним забарвленням фігур, що відображені, та імітацією різних світлових ефектів [4-7].

Для відображення функцій однієї змінної $y(x)$ використовують графіки в декартовій (прямокутній) системі координат. При цьому зазвичай будують дві осі – горизонтальну X і вертикальну Y , і задають координати x і y , які визначають вузлові точки функції $y(x)$. Оскільки MATLAB – матрична система, сукупність точок $y(x)$ задана векторами X і Y однакового розміру [4-7].

Команда **plot** (X , Y) служить для побудови графіків функцій у декартовій системі координат, координати точок (x, y) беруть із векторів однакового розміру X і Y . Якщо X або Y - матриця, то будується сімейство графіків за даними, що містяться в колонках матриці.

Наприклад, для побудови графіка функції **sin(x)** на інтервалі від 0 до 2π необхідно виконати програму з рис. 2.25. Програма побудує графік залежності, який відображений у вікні **Figure 1** (рис. 2.26).

Команда **plot**(X , Y , S) аналогічна команді **plot**(X , Y), але тип лінії графіка можна задавати за допомогою рядкової константи S .

```

>> t=0:pi/100:2*pi;
>> y=sin(t);
>> plot(t,y)
>>

```

Рис. 2.25. Програма побудови графіка функції *sin* в інтервалі від *0* до *2π*

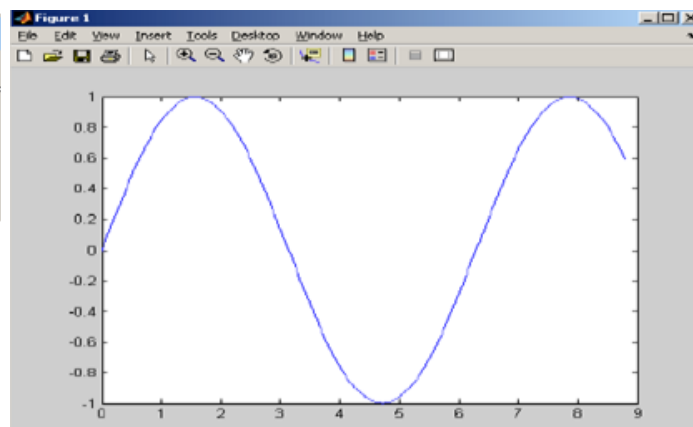


Рис. 2.26. Графік залежності $y = f(x)$, побудований функцією `plot(x, y)`

Значеннями константи *S* можуть бути символи, які подані в табл. 2.22 [4-7].

Таблиця 2.22

Задавання типу лінії

Колір лінії		Тип точки (стиль)		Тип лінії (маркер)	
y	жовтий	.	Точка	-	Суцільна
m	фіолетовий	o	Коло	:	Подвійний пунктир
c	блакитний	x	Хрест	-.	Штрих-пунктир
r	червоний	+	Плюс	--	штрихова
g	зелений	*	Зірочка		
b	синій	s	Квадрат		
w	білий	d	Ромб		
k	чорний	v	Трикутник (вниз)		
		^	Трикутник (вверх)		
		<	Трикутник (вправо)		
		>	Трикутник (вліво)		
		p	П'ятикутник		
		h	Шестикутник		

Отже, за допомогою рядкової константи S можна змінювати колір лінії, подавати вузлові точки різними відмітками (точка, коло, хрест, трикутник із різною орієнтацією вершини і т. д.) і змінювати тип лінії графіка.

Команда **plot**($X1, Y1, S1, X2, Y2, S2, X3, Y3, S3...$) будує на одному графіку ряд ліній, поданих даними вигляду (X, Y, S), де X і Y - вектори або матриці, а S - рядки. За допомогою такої конструкції можлива побудова, наприклад, графіка функції лінією, колір якої відрізняється від кольору вузлових точок. За відсутності вказівки на колір ліній і точок він вибраний автоматично з таблиці кольорів (білий виключено). Якщо ліній більше шести, то повторюють вибирати кольори.

Розглянемо приклад побудови графіків трьох функцій із різними стилями подання кожного з них $y1 = \sin(x)$; $y2 = \sin^2(x)$; $y3 = \sin^3(x)$ на інтервалі від -2π до 2π із кроком $0,1$ (рис. 2.27).

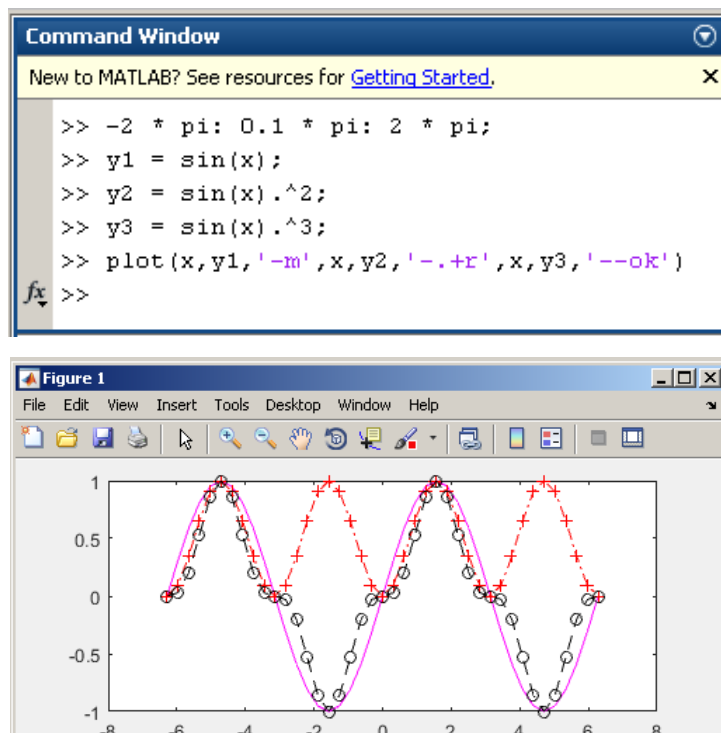


Рис. 2.27. Побудова графіків трьох функцій різними стилями

Із рисунка видно, що графік функції y_1 побудований суцільною лінією, графік y_2 - пунктирною лінією з маркерами у вигляді знака «плюс», а графік y_3 - штриховий лінією з кружечками.

Іноді потрібно порівняти поведінку двох функцій, значення яких сильно відрізняються один від одного. Графік функції з невеликими значеннями практично зливається з віссю абсцис, і встановити його вигляд не вдається. У цій ситуації допомагає функція **plotyy**, яка виводить графіки у вікно з двома вертикальними осями, що мають відповідний масштаб.

Для побудови графіка в полярній системі координат використовують функцію **polar** [4-7].

Розглянемо приклад побудови графіка функції $\sin(3\varphi)$ на інтервалі від 0 до 2π із кроком $0,001$ у полярній системі координат (рис. 2.28).

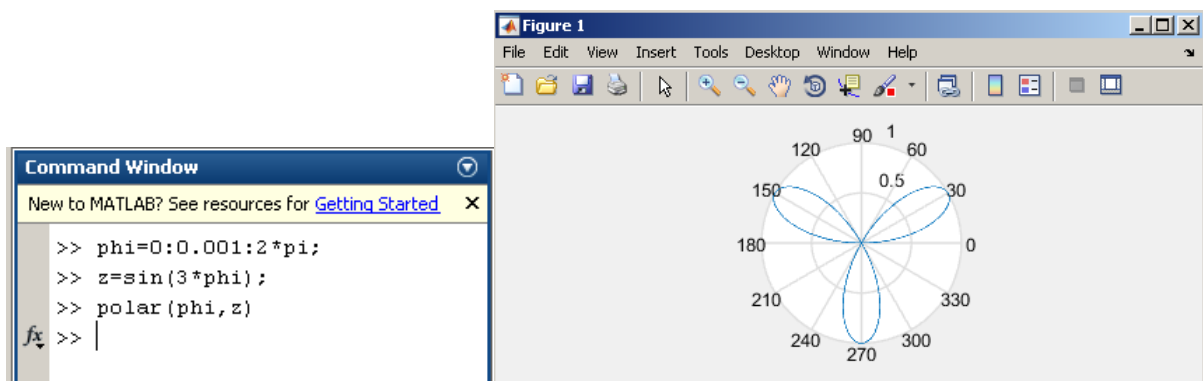


Рис. 2.28. Побудова графіка функції в полярній системі координат

Тривимірні поверхні звичайно описують функцією двох змінних $z(x, y)$. Специфіка побудови тривимірних графіків потребує не просто задавання ряду значень x і y , тобто векторів x і y , а визначення для X і Y двовимірних масивів - матриць.

Для створення таких масивів служить функція **meshgrid**. В основному її використовують спільно з функціями побудови графіків тривимірних поверхонь. Функцію **meshgrid** записують у таких формах:

- $[X,Y,Z] = \text{meshgrid}(x,y,z)$ - повертає тривимірні масиви, що використовують для обчислення функцій трьох змінних і побудови тривимірних графіків;

- $[X,Y] = \text{meshgrid}(x,y)$ - перетворить область, задану векторами x і y , у масиви X і Y , які можуть бути використані для обчислення функції двох змінних і побудови тривимірних графіків. Рядки вихідного масиву X є копіями вектора x , а стовпці Y - копіями вектора y [4-7].

Команда **plot3(...)** є аналогом команди **plot(...)**, але стосується функції двох змінних $z(x, y)$. Вона будує аксонометричне зображення тривимірних поверхонь і подана такими формами:

- **plot3**(x, y, z) - будує масив крапок, поданих як вектори x, y і z , сполучаючи їх відрізками прямих. Ця команда має обмежене вживання;

- **plot3**(X, Y, Z, S) - забезпечує побудову зі специфікацією стилю ліній і точок;

- **plot3**($x1, y1, z1, s1, x2, y2, z2, s2, \dots$) - будує на одному графічному полі графіки декількох функцій $z1(x1,y1), z2(x2,y2)$ і т. д. із специфікацією ліній і маркерів кожної з них [4-7].

Найбільш поширеними і наочними є сітчасті графіки поверхонь із заданим або функціональним забарвленням. У назві їхніх команд присутнє слово **mesh**. Є дві групи таких команд:

- **mesh**(X, Y, Z, C) - виводить у графічне вікно сітчасту поверхню $Z(X,Y)$ із кольорами вузлів поверхні, заданими масивом C ;

- **mesh**(X, Y, Z) - аналог попередньої команди, якщо $C = Z$.

На рис. 2.29 наведений приклад використання команди **mesh**.

2.4.2. Форматування графіків

Після того як графік уже побудований, MATLAB дає змогу форматувати або оформити його в потрібному вигляді. Так, для установлення над графіком титульного підпису використовують команду **title** ('*підпис*').

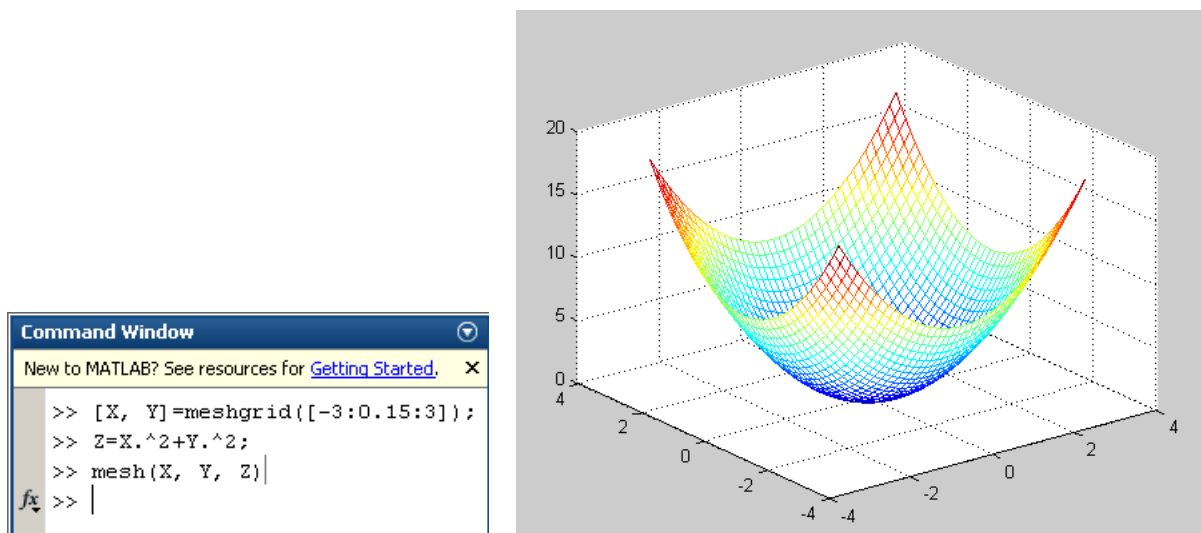


Рис. 2.29. Графік поверхні, створений командою `mesh(X,Y,Z)`

Для установлення підписів біля осей x , y і z використовують такі команди: `xlabel('підпис')`, `ylabel('підпис')`, `zlabel('підпис')` (рис. 2.30).

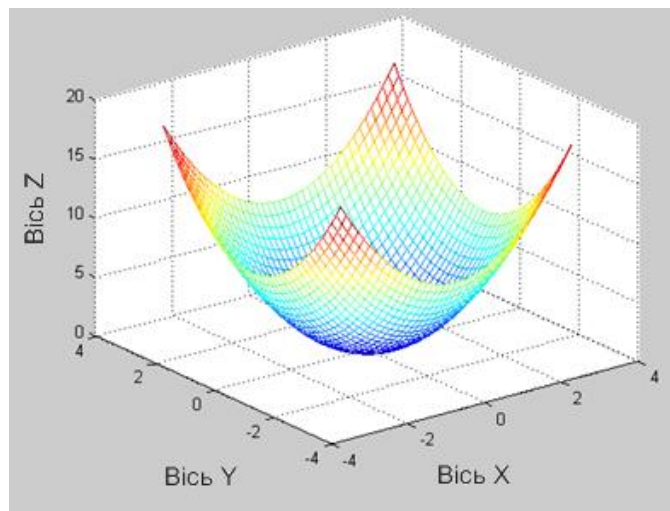


Рис. 2.30. Установлення написів з використанням команд `xlabel('Вісь X')`, `ylabel('Вісь Y')`, `zlabel('Вісь Z')`

Часто виникає необхідність додавання тексту в певне місце графіка, наприклад для позначення тієї або іншої кривої графіка. Для цього використовують команду **text**:

- **text**(X,Y, 'string') – додає у двовимірний графік текст, заданий рядковою константою 'string', так що початок тексту розташований у точці з координатами (X, Y). Якщо X і Y задані як одновимірні масиви, то напис поміщений у всі позиції $[x(i), y(i)]$;

- **text**(X,Y, Z, 'string') – додає в тривимірний графік текст, заданий рядковою константою 'string', так що початок тексту розташований у позиції, заданій координатами X, Y і Z [4-7].

Дуже зручний спосіб введення тексту надає команда **gtext**:

- **gtext**('string') – задає текст, що виводять на графік, у вигляді рядкової константи 'string' і виводить на графік маркер у вигляді хрестика, який переміщують за допомогою миші. Встановивши маркер у потрібне місце, достатньо натиснути будь-якою кнопкою миші для виведення тексту.

Пояснення у вигляді відрізків ліній із довідковими написами, що розміщені всередині графіка або біля нього, називають легендою. Для створення легенди використовують різні варіанти команди **legend**:

legend(string1, string2,..., strings) – додає до поточного графіка легенду у вигляді рядків, вказаних у списку параметрів (рис. 2.31);

legend (Pos) – поміщає легенду у визначене місце, специфіковане параметром Pos:

Pos=0 – краще місце, вибрано автоматично;

Pos=1 – верхній правий кут;

Pos=2 – верхній лівий кут;

Pos=3 – нижній лівий кут;

Pos=4 – нижній правий кут;

Pos=-1 – праворуч від графіка [4-7].

Додаючи легенду, потрібно врахувати, що порядок і кількість аргументів команди **legend** мають відповідати порядку виведення графіків і їхній кількості.

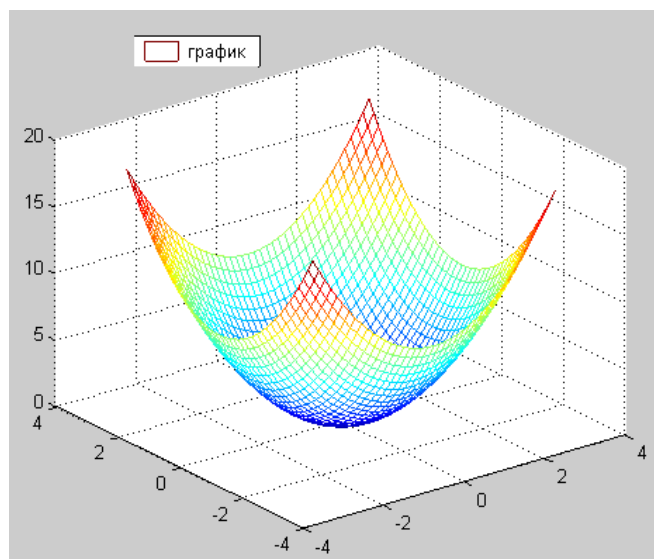


Рис. 2.31. Графік із поясненнями **legend('график')**

Звичайно графіки виводять у режимі автоматичного масштабування. Розглянемо команди класу **axis**, що змінюють цю ситуацію:

- **axis**([XMIN XMAX YMIN YMAX]) – установлення діапазонів координат по осях x і y для поточного двовимірного графіка;
- **axis**([XMIN XMAX YMIN YMAX ZMIN ZMAX]) – установлення діапазонів координат по осях x , y і z поточного тривимірного графіка;
- **axis auto** - установлення параметрів осей за замовчуванням.

Будуючи графіки, додатково для розмітки осей часто використовують масштабну сітку. Команди **grid** дають змогу задавати побудову сітки або відмінити цю побудову:

- **grid on** – додає сітку до поточного графіка;
- **grid off** – відключає сітку [4-7].

У багатьох випадках необхідна побудова декількох накладених один на одного графіків в одному і тому самому вікні. Для цього служить команда продовження графічних побудов **hold**. Її використовують у таких формах:

- **hold on** – забезпечує продовження виведення графіків у поточне вікно, щоб додавати подальші графіки до вже існуючих;
- **hold off** – відміняє режим продовження графічних побудов.

2.4.3. Розбиття графічного вікна на декілька вікон для побудови графіків

Буває, що в одному вікні потрібно розташувати декілька координатних осей із різними графіками без накладення їх один на одного. Для цього використовують команди **subplot**, що вказують перед побудовою графіків:

- **subplot(m, n, p)** - розбиває графічне вікно на $m \times n$ підвікон, при цьому m – кількість підвікон по горизонталі, n – кількість підвікон по вертикалі, а p – номер підвікна, у яке виводитимуть поточний графік (підвікна нумерують послідовно по рядках).

Проілюструємо роботу функції **subplot** (рис. 2.32).

Було сформовано три рядки і два стовпці полів для виведення графіків. Звернення до кожного конкретного поля відбувається із зазначенням його номера. Нумерація відбувається зліва направо і зверху до низу [4-7].

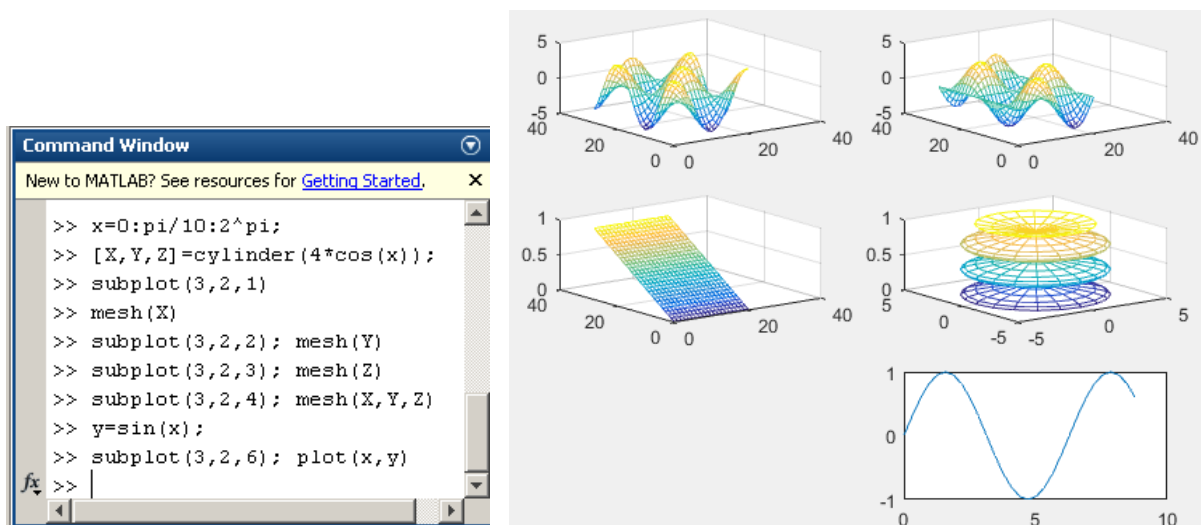


Рис. 2.32. Робота функції subplot

2.4.4. Побудова анімованих графіків

MATLAB дає змогу будувати анімовані графіки, що відображають траєкторію руху точки на площині або у просторі впродовж заданого інтервалу часу.

Для введення траєкторії руху точки задають закон змінювання її координат: $x(t)$, $y(t)$ – у випадку руху точки на площині; $x(t)$, $y(t)$ і $z(t)$ – у випадку руху точки у просторі. Параметр t є вектором-рядком, що містить набір значень з інтервалу часу, впродовж якого рухається точка [4-7].

Команди виведення траєкторії руху точки в MATLAB є такими:

- *comet*(x,y) – у двовимірному просторі;
- *comet3*(x,y,z) – у тривимірному просторі.

Приклад побудови траєкторії руху точки впродовж 10 с, координати якої змінюються за законом: $x(t) = \frac{\sin t}{t+1}$, $y(t) = \frac{\cos t}{t+1}$, показано на рис. 2.33.

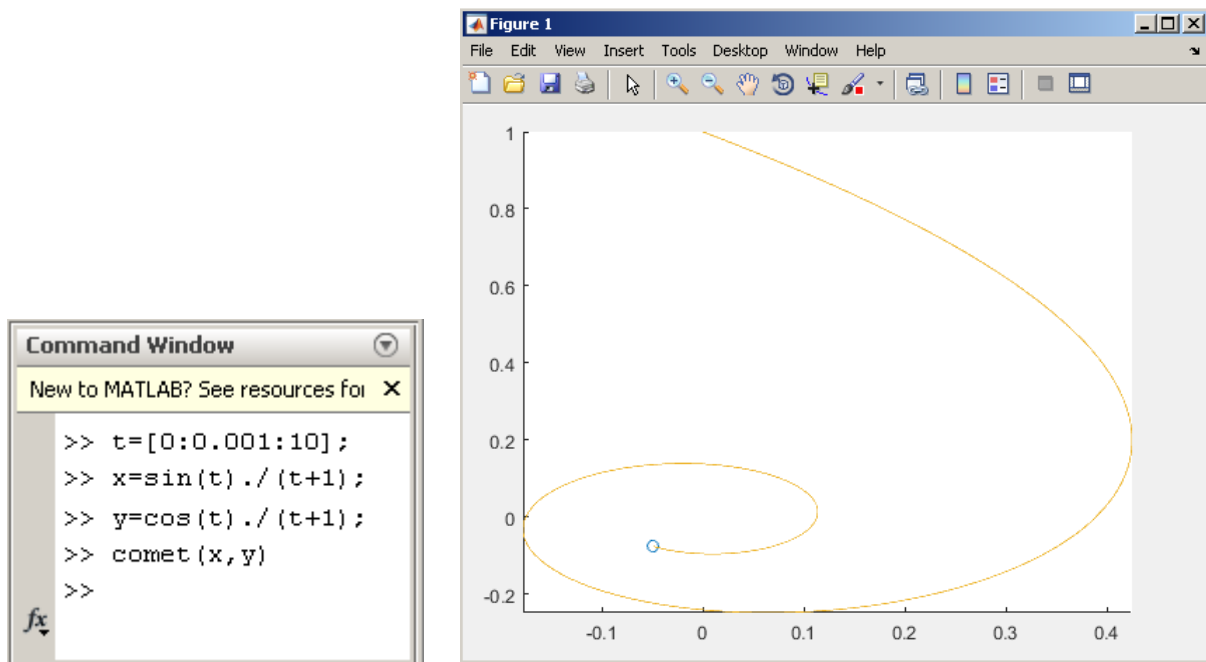


Рис. 2.33. Зображення руху точки у просторі за законом

$$x(t) = \frac{\sin t}{t+1}, \quad y(t) = \frac{\cos t}{t+1}$$

2.4.5. Побудова об'ємних поверхневих діаграм

У MATLAB для побудови об'ємних поверхневих діаграм використовують функцію SURF, SURFC - затінена сітчаста поверхня.

Синтаксис:

`surf (X, Y, Z, C) surfc (X, Y, Z, C);`

`surf (x, y, Z, C) surfc (x, y, Z, C);`

`surf (Z, C) surfc (Z, C).`

Опис: команда **surf (X, Y, Z, C)** виводить на екран сітчасту поверхню для значень масиву *Z*, визначених на множині значень масивів *X* і *Y*. Колір комірки визначає масив *C*.

Команда **surf (x, y, Z, C)** виконує ту саму функцію, але замість двовимірних масивів *X*, *Y* використовує їхні одновимірні проєкції, так що якщо $\text{length}(x) = n$, а $\text{length}(y) = m$, то $[m, n] = \text{size}(Z)$. У цьому випадку вузли сітчастої поверхні визначені трійками $\{x(j), y(i), Z(i, j)\}$, де вектор *x* визначає стовпці масиву *Z*, а *y* – рядки.

Команда **surf (Z, C)** використовує сітку, яка визначена одновимірними масивами $x = 1:n$ і $y = 1:m$ [4-7].

Група команд **surfc (...)** на додачу до тривимірних затінених поверхонь будує проєкцію ліній постійного рівня.

Приклади:

1. Побудуємо тривимірну затінену поверхню функції $z = x * \exp(-x^2 - y^2)$ (рис. 2.34).

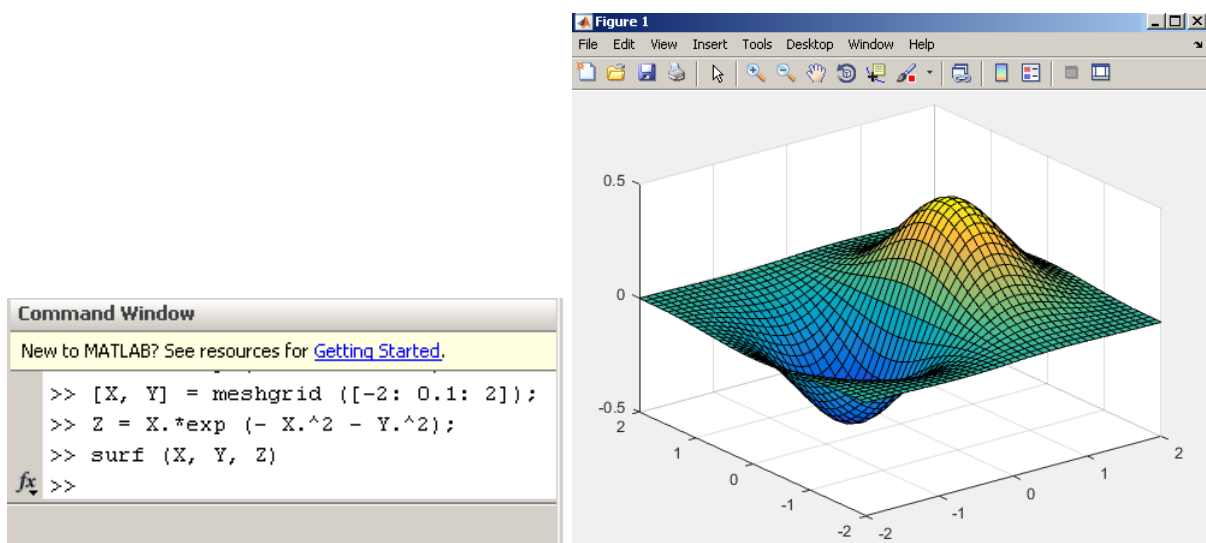


Рис. 2.34. Графік тривимірної поверхні $z = x * \exp(-x^2 - y^2)$

2. Розглянемо приклад сфери, яка розфарбована відповідно до матриці Адамара (Hadamard), часто використовуваної в теорії кодування сигналів і складеної тільки з двох чисел 1 і -1 (рис. 2.35).

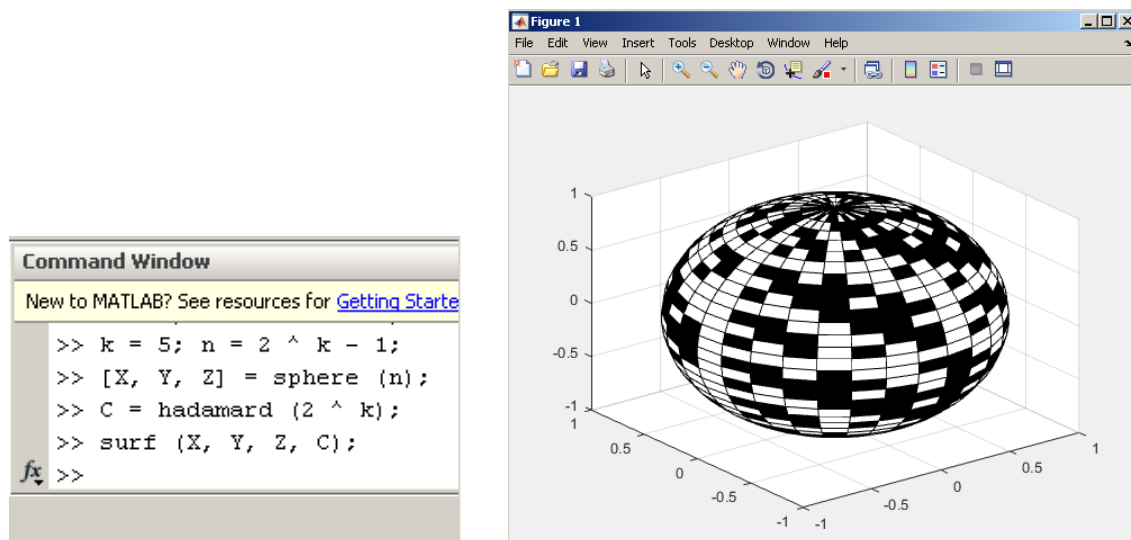


Рис. 2.35. Графік сфери, яка розфарбована відповідно до матриці Адамара

ЗАВДАННЯ ДЛЯ ВИКОНАННЯ

Завдання 1. Побудувати графіки функцій f_1, f_2, f_3 згідно із завданням (табл. 2.23). Вихідні дані задавати самому. Виводити графіки в одному графічному вікні, графіки мають бути підписані, відмасштабовані.

Таблиця 2.23

Варіанти завдання 1

Варіант	f_1	f_2	f_3	f_4
1	2	3	4	5
1	$\sin(x)$	$\cos(x)$	x^2	$\cos(r)/r$
2	e^x	x^2	x	$\cos^2(r)/r$
3	$\sin(x) + \cos(x)$	$\cos(x) + x^2$	$x^2 + \lg(x)$	$\cos(r^2)/r$
4	$\sin(x) + e^x$	$\sin(x) + x^2$	$\sin(x) + x$	$\cos(r)/r^2$
5	$x * \sin(x)$	$x * \cos(x)$	x^2	$(\cos(r)/r)^2$

Продовження табл. 2.23

1	2	3	4	5
6	xe^x	$\sin(x) + x^2$	$\sin(x) + x$	$\sin^2(r)/r$
7	$\sin(x) * \cos(x)$	$\cos(x) * x^2$	$x^2 \lg(x)$	$\sin(r^2)/r$
8	$\sin(x)e^x$	$\sin(x) * x^2$	$\sin(x) * x$	$\sin(r)/r^2$
9	$\sin^2(x)$	$\cos^2(x)$	x^3	$(\sin(r)/r)^2$
10	$\sin(x) * e^x$	$\sin(x) * x^2$	$\sin(x) * x$	$r + \cos(r)/r$

Завдання 2. Виконати наступний варіант завдання 1, але графіки функцій будувати кожний у своєму підвікні графічного вікна.

Завдання 3. Побудувати полярний графік функції. Варіант вибрати з табл. 2.24. Відформатувати отриманий графік.

Таблиця 2.24

Варіанти завдання 3

Варіант	Функція	Варіант	Функція
1	$\rho = 2 + 3 \sin \varphi$	2	$\rho = 4 \cos 2 \left(\varphi + \frac{\pi}{4} \right)$
3	$\rho = 2 + \sin 2 \left(\varphi + \frac{\pi}{4} \right)$	4	$\rho = 3 - 2 \cos 3\varphi$
5	$\rho = 2 - 6 \cos 3\varphi$	6	$\rho = 2 + 4 \cos \frac{\varphi}{2}$
7	$\rho = 3 + 2 \sin 3 \left(\varphi + \frac{\pi}{2} \right)$	8	$\rho = 3 \sin 3 \left(\varphi + \frac{\pi}{6} \right)$
9	$\rho = 2 + 3 \sin \frac{\varphi}{4}$	10	$\rho = 3 - 2 \cos 3\varphi$

Завдання 4. Побудувати графік тривимірної поверхні для функції f_4 з табл. 2.23, де $r = \sqrt{x^2 + y^2}$). Вхідні дані задати самому.

Завдання 5. Побудувати графік руху точки в просторі. Завдання взяти із табл. 2.23 для $x = 0,1$ до 2π із кроком $0,001$, а $y = f_1/f_3$.

Завдання 6. Створити більш складний анімований графік – коливання струни. Для цього виконати команди

```
>>x=0:0.01:1;
>>for n=0:50
    plot(x,sin(n*pi/5)*sin(pi*x)),axis([0,1,-2,2])
    m(n+1)=getframe;
end
>>movieview(m)
```

Що потрібно зробити, щоб збільшити амплітуду коливання? Зробити відповідну зміну.

Завдання 7. За допомогою оператора surf (...) побудувати поверхню, задану функцією $z(x, y)$ (табл. 2.25). Використати поелементні операції.

Таблиця 2.25

Варіанти для завдання 7

Варіант	Опис поверхні	Варіант	Опис поверхні
1	u=[0:0.05:pi]'; v=[0:0.05:pi]; X=cos(u)*cos(v); Y=cos(u)*sin((v)); Z=sin(u)*ones(size(v));	2	[x,y]=meshgrid(-5:0.2:5); a=20; b=20; c=1; A=a ² ; B=b ² ; z=c*sqrt(x ² /A+y ² /B+1);
3	[x,y]=meshgrid(-2*pi:0.1*pi:2*pi); a=2; b=2; c=1; A=a ² ; B=b ² ; z=c*sqrt(x ² /A+y ² /B);	4	[x,y]=meshgrid(-5:0.2:5); z=sqrt(100-x ² -y ²);
5	u=[-pi:0.02*pi:pi]'; v=[-pi:0.02*pi:pi]; X=sin ² u *sin v ² ; Y=0.9*(cos ² u*sin ² v; Z=sin u*ones(size(v))	6	[x,y]=meshgrid(-5:0.2:5); a=20; b=20; c=1; A=a ⁵ ; B=b ⁵ ; z=0.5*x ⁵ /A-y ⁵ /(B+1);
7	u=[-2*pi:0.05*pi:2*pi]'; v=[-2*pi:0.05*pi:2*pi]; X=u ² *cos(v); Y=u*sin(v); Z=u*ones(size(v));	8	u=[-2*pi:0.1*pi:2*pi]'; v=[-2*pi:0.2*pi:2*pi]; X=cos(u*v); Y=sin(u*v); Z=u*ones(size(v));
9	u=[0:0.025*pi:2*pi]'; v=[0:0.025*pi:2*pi]; X=sin(u)*cos(v); Y=cos(u)*sin(v); Z=sin(u)*ones(size(v));	10	u=[0:0.025*pi:2*pi]'; v=[0:0.025*pi:2*pi]; X=erf(u)*sin(v); Y=erf(u)*cos(v); Z=sin(u)*ones(size(v));

2.5. Програмування в середовищі MATLAB

Ознайомлення з елементами програмування в MATLAB, створенням і використанням М-файлів, операціями відношення, логічними операціями, умовними операторами і операторами циклу, набуття навичок їх використання для обчислень [4-7].

2.5.1. Редактор М-файлів

Найзручнішим способом виконання груп команд MATLAB є використання М-файлів, у яких можна набирати команди, виконувати їх усі одразу або частинами, зберігати у файлі і використовувати в подальшому. Для роботи з М-файлами призначений редактор *М-файлів*. За його допомогою можна створювати власні функції і викликати їх, зокрема, і з командного вікна [4-7].

Розрізняють два типи М-файлів: файл-програми (Script M-Files) і файл-функції (Function M-Files).

Для створення і редагування М-файлів призначений багатовіконний вмонтований редактор-відладчик **Editor** (Редактор), який можна викликати командою **edit** із командного рядка або командою меню **File** → **New** → **M-file**, або комбінацією клавіш **Ctrl** + **N** (рис. 2.36).

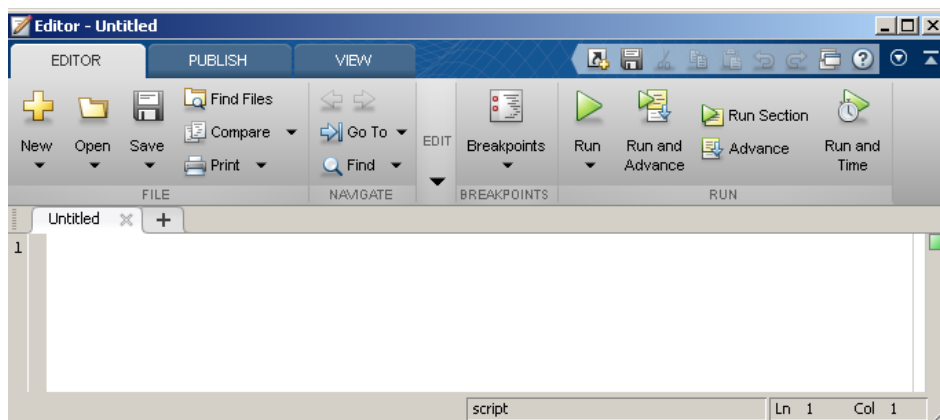


Рис. 2.36. Багатовіконний редактор-відладчик

Вигляд рядка і панелі інструментів залежать від версії MathCad і ширини вікна. Для прикладу наберемо програму, як на рис. 2.37.

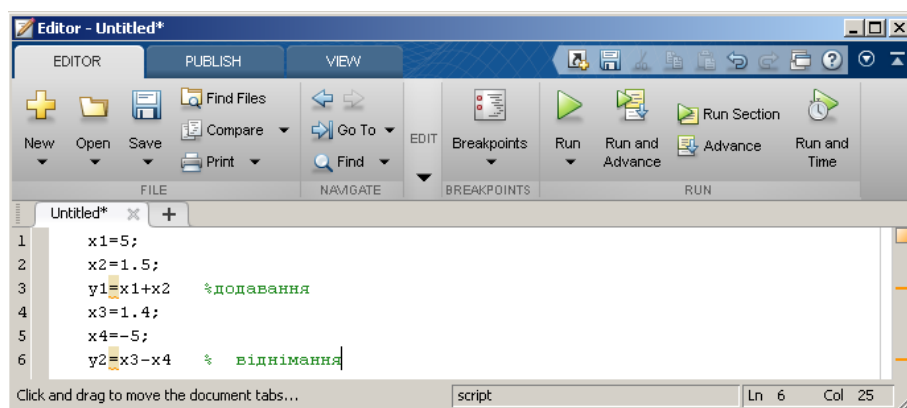


Рис. 2.37. Програма у вікні редактора

Перед запуском програми її потрібно зберегти на магнітному диску (команда головного меню **File / Save As**) під ім'ям, наприклад, **Prim1.m**. Запускаючи виконання всіх команд, що містяться у файлі, потрібно вибрати

пункт **Run** у меню **RUN** (піктограма  на панелі інструментів) або натиснути клавішу **F5**. Але робити попереднє збереження як окрему операцію не обов'язково. Допустимо одразу розкрити меню **Debug** (або іншим способом), у якому опцію **Run** у цьому випадку заміняє опція **Save and Run**, що дає змогу запустити програму, попередньо зберігши її автоматично.

Для запуску файлу на виконання його необхідно записати на диск, використовуючи команду **Save as** у меню **File** редактора. Редактор-відладчик *m*-файлів виконує синтаксичну перевірку програмного коду з введенням тексту. При цьому використано таке колірне виділення:

- ключові слова мови програмування – синій колір;
- оператори, константи і змінні – чорний колір;
- коментарі після знака «%» – зелений колір;

- символні змінні (в апострофах) – коричневий колір;
- синтаксичні помилки – червоний колір.

Завдяки колірному виділенню вірогідність синтаксичних помилок різко знижується [4-7].

Результат виконання прикладу наведений на рис. 2.38 у командному вікні.

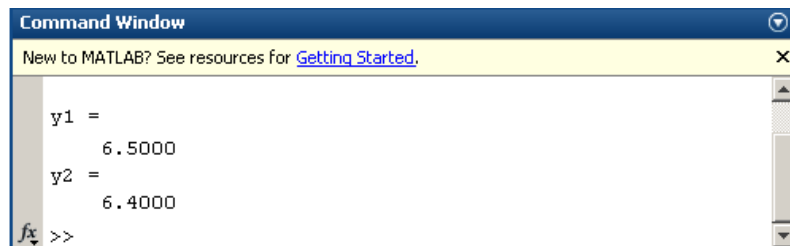


Рис. 2.38. Результат виконання програми

М-файли, які створює редактор-відладчик, поділяють на два класи: **файли-сценарії**, що не мають вхідних параметрів, і **файли-функції**, що мають вхідні параметри.

2.5.2. Файл-сценарій (називають також **script**-файлом) є просто записом серії команд без вхідних і вихідних параметрів. Він має таку структуру:

% Основний коментар

% Додатковий коментар

Тіло файлу з будь-якими виразами.

Важливі такі властивості файлів - сценаріїв:

- 1) вони не мають вхідних і вихідних аргументів;
- 2) працюють із даними з робочої області;
- 3) у процесі виконання не компільовані;
- 4) є зафіксованою у вигляді файлу послідовністю операцій, повністю аналогічною тій, що використана в сесії.

Розглянемо файл-сценарій, наведений на рис. 2.39.

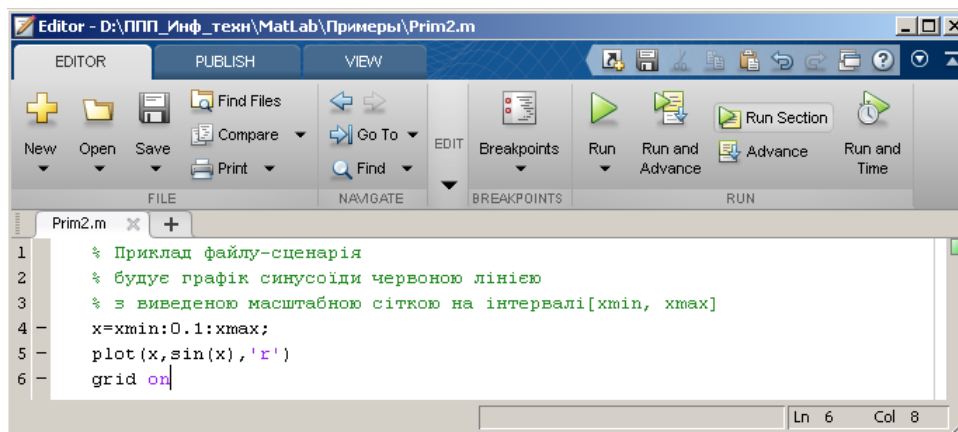


Рис. 2.39. Створення файлу-сценарію в MATLAB

Перші три рядки тут – це коментар, інші – тіло файлу. Зверніть увагу на можливість задавання коментаря російською мовою. Знак «%» у коментарях має починатися з першої позиції рядка. Слід зазначити, що такий файл не можна запустити без попередньої підготовки, що зведено до задавання значень змінним *xmin* і *xmax*, які використані в тілі файлу (рис. 2.40). Це впливає з першої властивості файлів-сценаріїв – вони працюють із даними з робочої області. Імена файлів-сценаріїв не можна використовувати як параметри функцій, оскільки файли-сценарії не повертають значень. Можна сказати, що файл-сценарій – це найпростіша програма мовою програмування MATLAB [4-7].

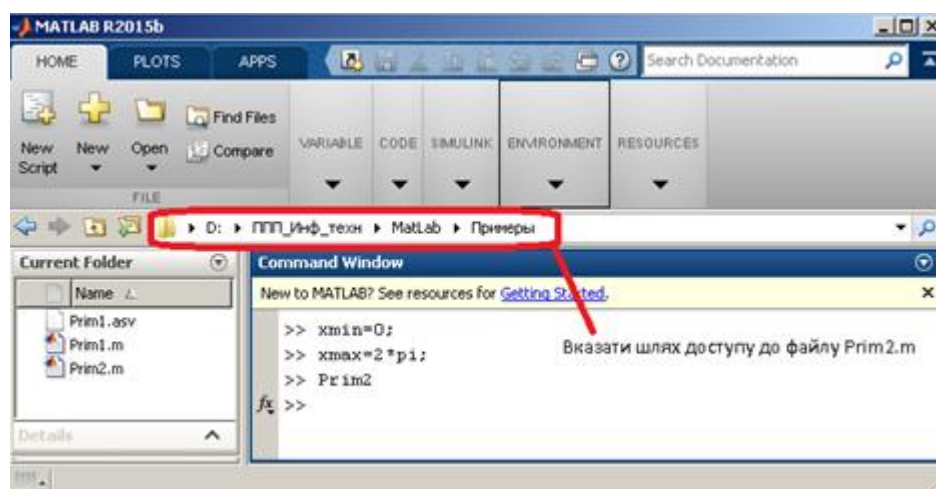


Рис. 2.40. Виклик файлу-сценарію з командного вікна

Результат виведений (для нашого прикладу) в графічному вікні, як показано на рис. 2.41.

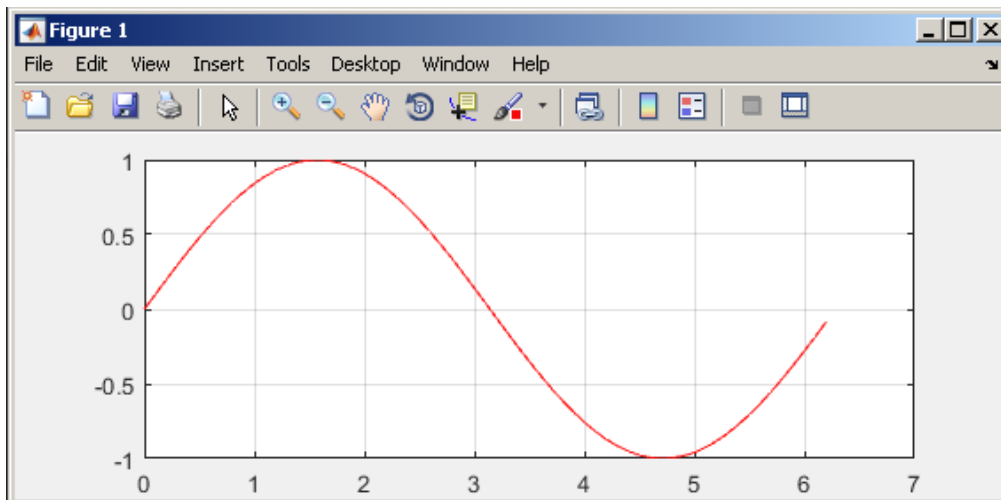


Рис. 2.41. Отриманий результат у графічному вікні

2.5.3. Файл-функція є типовим об'єктом мови програмування системи MATLAB. Одночасно він є повноцінним модулем для структурного програмування, оскільки містить вхідні і вихідні параметри і використовує апарат локальних змінних. Структура такого модуля з одним вихідним параметром має такий вигляд:

function var = f_name(Список_параметрів)

% Основний коментар

% Додатковий коментар

Тіло файлу з будь-якими виразами

var=вираз.

Приклад функції обчислення $f = e^{-x} \sqrt{\frac{x^2+1}{x^4+0.1}}$ наведено на рис. 2.42.

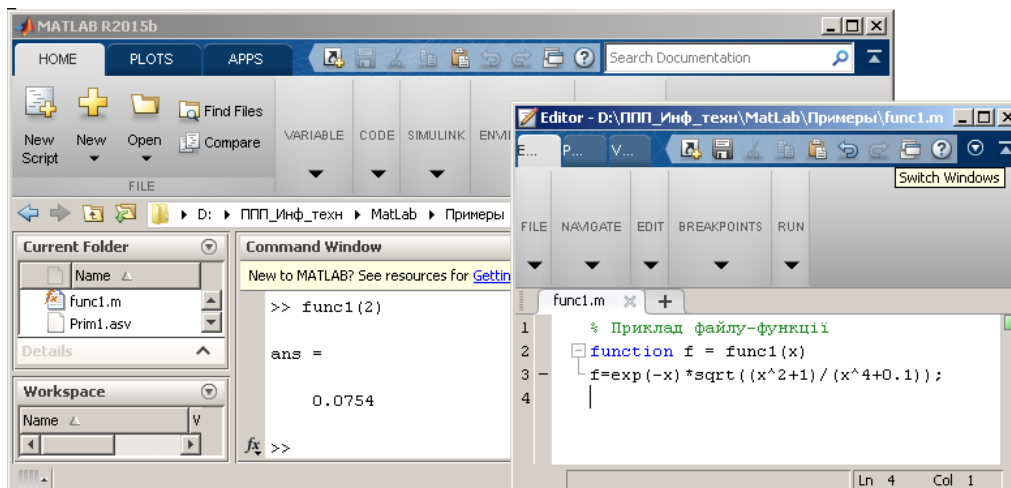


Рис. 2.42. Приклад використання файлу-функції

M-файл-функція має такі властивості:

- 1) починається з оголошення `function`, після якого вказано ім'я змінної `var` - вихідного параметра, ім'я самої функції `f_name` і список її вхідних параметрів;
- 2) функція повертає своє значення і може бути використана в математичних виразах;
- 3) усі змінні, що є в тілі файлу-функції, є локальними, тобто діють тільки в межах тіла функції;
- 4) файл-функція є самостійним програмним модулем, який обмінюється з іншими модулями через свої вхідні і вихідні параметри;
- 5) правила виведення коментарів такі самі, що і для файлів-сценаріїв;
- 6) із виявленням файлу-функції він компілює себе і потім виконуваний, а створені машинні коди зберігаються в робочій області системи MATLAB [4-7].

Остання конструкція `var = вираз` введена, якщо потрібно, щоб функція повертала результат обчислень. Наведена форма файлу-функції характерна для функції з одним вихідним параметром. Якщо вихідних параметрів більше, то вони вказані у квадратних дужках після слова `function`. При цьому структура модуля має такий вигляд:

function [var1,var2....] = **f_name**(Список_параметрів)

% Основний коментар

% Додатковий коментар

Тіло файлу з будь-якими виразами

var1=вираз

var2=вираз.

Таку функцію використовують як окремий елемент програм вигляду [var1,var2]=f_name(*Список_параметрів*). Після його вживання змінні виходу var1, var2 стають визначеними, і їх можна використовувати в подальших математичних виразах та інших сегментах програми.

2.5.4. Оператори мови MATLAB

2.5.4.1. Оператори введення-виведення. Для організації діалогового введення і виведення використовують оператори, подані в табл. 2.26.

Таблиця 2.26

Оператори діалогового введення-виведення

Оператор	Синтаксис	Призначення
INPUT	$x = \text{input}('<\text{запрошення}>')$	Для введення даних із клавіатури
DISP	$\text{disp} (<\text{змінна або текст в апострофах}>)$	Для виведення на дисплей

Наведемо простий приклад діалогової програми, яка служить для багаторазового обчислення довжини кола за значенням радіуса r , що вводить користувач (рис. 2.43).

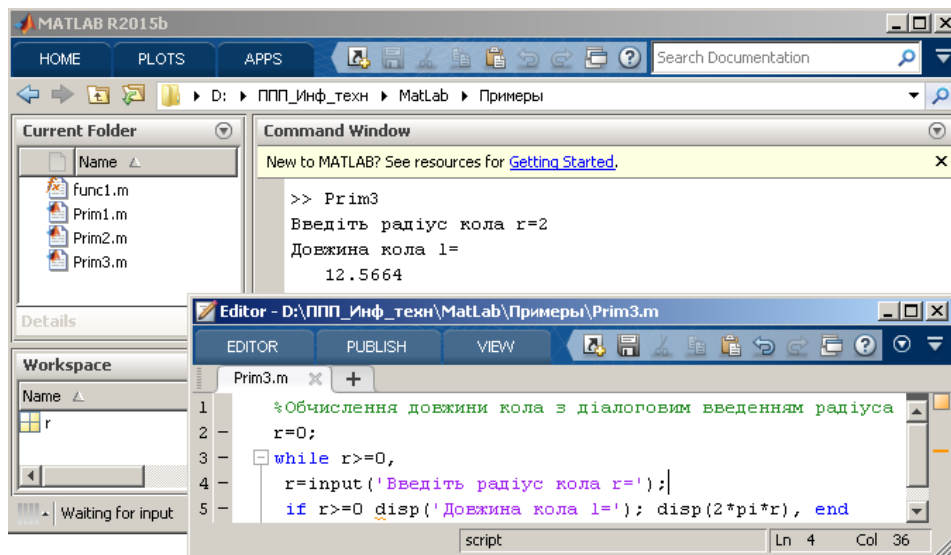


Рис. 2.43. Приклад діалогової програми

2.5.4.2. Умовні оператори. Для організації розгалужень служать *умовні оператори*.

Конструкції умовних операторів:

- | | |
|---|---|
| <p>1) if <умова>
 <оператори>
End;</p> | <p>2) if <умова>
 <оператори 1>
else
 <оператори 2>
end.</p> |
|---|---|

Оператори (тіло виразу) виконувані тільки в тому випадку, якщо умова істинна (буде виконана), якщо умова не виконувана, то тіло виразу пропущено (не виконуване).

Якщо хід програми має змінюватися залежно від декількох умов, то слід використовувати повну конструкцію **if-elseif-else**. Кожна з гілок **elseif** у цьому випадку має містити умову виконання блока операторів, розміщених після неї. Важливо розуміти, що умови перевіряють підряд, перша виконана умова призводить до роботи відповідного блока, виходу з конструкції **if-elseif-else** і переходу до оператора, наступного за **end**. В останній гілці **else** не має бути ніякої умови. Оператори, що знаходяться між **else** і **end**, працюють у тому випадку, якщо всі умови виявилися невиконаними.

Наприклад, написати файл-функцію для обчислення кусково-заданої функції (рис. 2.44)

$$f(x) = \begin{cases} 1 - e^{-1-x}, & x < -1; \\ x^2 - x - 2, & -1 \leq x \leq 2; \\ 2 - x, & x > 2. \end{cases}$$

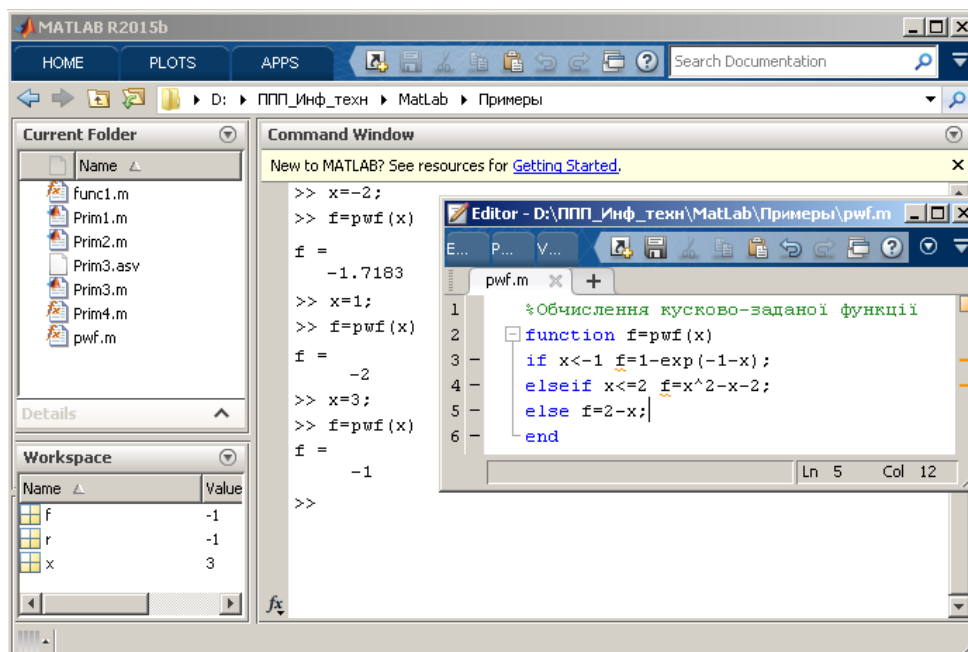


Рис. 2.44. Лістинг програми для обчислення значення кусково-заданої функції

2.5.4.3. Логічні оператори (оператори порівняння). У системі MATLAB можуть бути застосовані оператори порівняння (табл. 2.27).

Таблиця 2.27

Оператори порівняння

Символ	Призначення	Ім'я функції
<	Менше	lt
>=	Більше або дорівнює	ge
>	Більше	gt
<=	Менше або дорівнює	le
==	Дорівнює	eq
~=	Не дорівнює	ne

Оператори ($=$ і $\sim$) порівнюють дійсні та уявні частини комплексних чисел (зверніть увагу на два знаки дорівнює), а операції ($>$ $>=$ $<=$) - тільки дійсні частини.

Логічні операції можна записувати у вигляді функцій (табл. 2.28).

Таблиця 2.28

Логічні операції

Символ	Призначення	Ім'я функції
&	Логічне «і»	and
 	Логічне «або»	or
~	Заперечення	not

Результатом логічних операцій є числа **0** (*false*) і **1** (*true*).

2.5.4.4. Оператори циклу. У системі MATLAB є два різновиди операторів циклу - умовний і арифметичний. Для повторення операторів нефіксовану кількість раз використано оператор циклу з передумовою:

Оператор циклу while

```
while <умова>
    <оператори>
end.
```

Оператори виконуватимуться, якщо змінна *<умова>* має ненульові елементи.

Розглянемо приклад обчислення суми. Потрібно знайти суму ряду для заданого x (розкладання в ряд ***sin (x)***):

$$S(x) = \sum_{k=0}^{\infty} (-1)^k \frac{x^{2k+1}}{(2k+1)!}.$$

Суму можна накопичувати до тих пір, поки складові не стають занадто маленькими, скажімо більше за модулем 10^{-10} . Розв'язуємо задачу із застосуванням циклу **while**, який працює, поки виконувана умова циклу (рис. 2.45).

Арифметичний оператор циклу має такий вигляд:

```

for <ім'я> = <Пз>: <Крок>: <КЗ>
    <оператори>
end,

```

де <ім'я> – ім'я параметра циклу (змінна, яка управляє циклом);

<Пз> – початкове значення параметра циклу;

<КЗ> – кінцеве значення параметра циклу;

<Крок> – приріст значень змінної < ім'я> із її зміною від значення <Пз> до значення <КЗ>. Якщо параметр <Крок> не вказаний, за замовчуванням його значення приймають рівним одиниці.

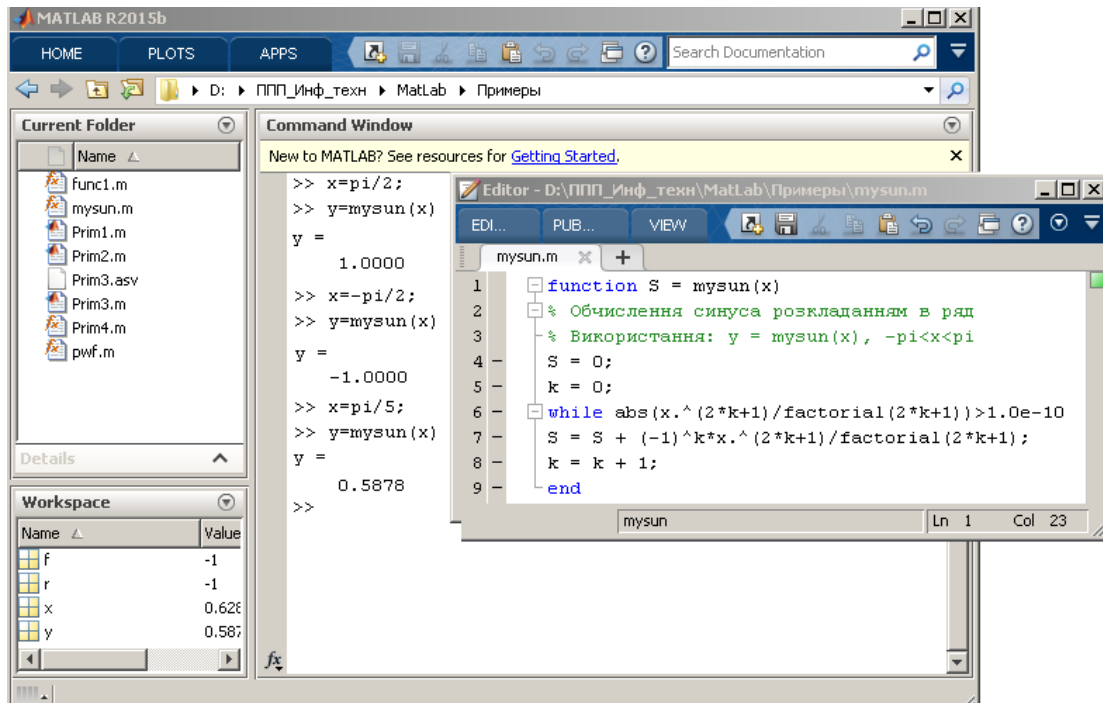


Рис. 2.45. Лістинг файл-функції `mysun`, яка обчислює синус розкладанням у ряд

Працюючи з циклом **for**, допустимо використовувати оператор переривання циклу **break**, працюючи з яким робота циклу завершується, і управління передано на наступний після кінця циклу оператор.

Наприклад, написати файл-програму для обчислення суми $S = \sum_{k=1}^{10} \frac{1}{k!}$.

Алгоритм обчислення суми використовує накопичення результату, тобто спочатку сума дорівнює нулю ($S = 0$), потім у змінну k заносять одиницю, обчислюють $1/k!$, додають до S і результат знову заносять у S . Далі k збільшується на одиницю, і процес триває, поки останньою складовою не стане $1/10!$. Файл-програма **Prim4** наведено на рис. 2.46.

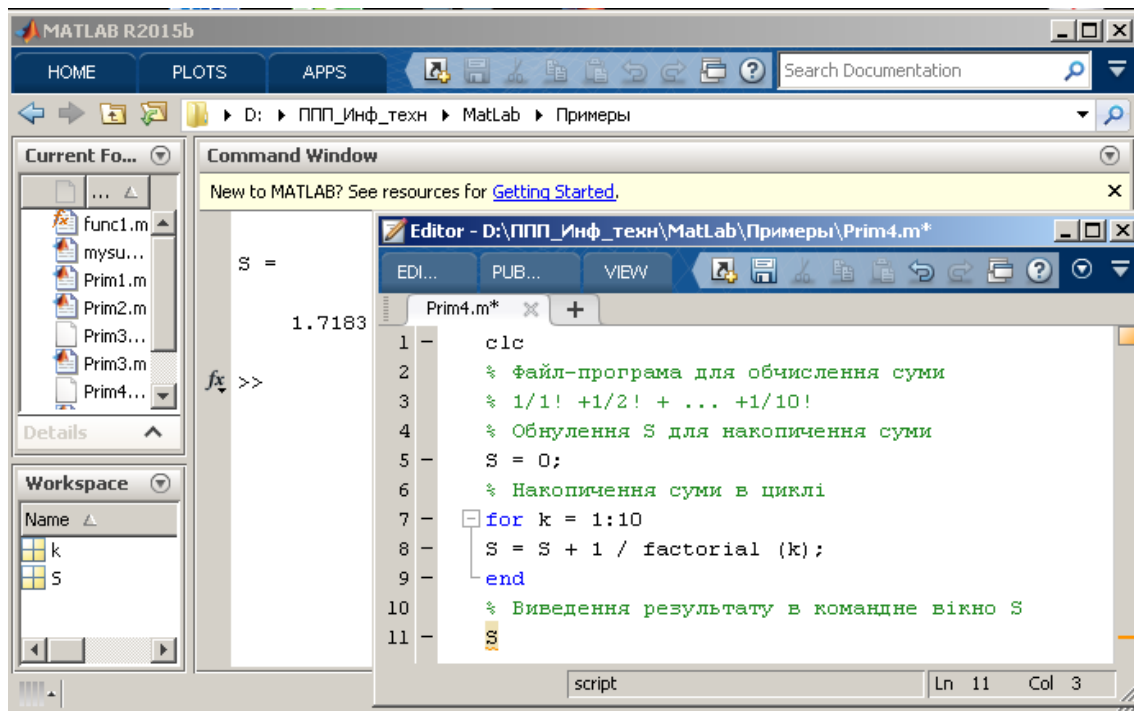


Рис. 2.46. Лістинг файл-програми Prim4 для обчислення суми

Спочатку необхідно набрати файл-програму в редакторі М-файлів, зберегти її в поточному каталозі у файлі **Prim4.m** і виконати. Результат буде відображений у командному вікні, оскільки в останньому рядку файл-програми `S` міститься без точки з комою для виведення значення змінної `S`.

Оператор циклу **while** у MATLAB використовують для багаторазового виконання блока коду, поки задана логічна умова залишається істинною (не нульовою) (табл. 2.29). Цикл перевіряє умову перед кожною ітерацією; якщо вона хибна - код завершується. Завжди завершується ключовим словом `end`.

Формат оператора while	Приклад. Обчислити суму чисел від 1 до 5
while умова % оператори end	<pre>i = 1; S = 0; while i <= 5 S = S + i; i = i + 1; end disp(S)</pre>

2.5.4.5. Оператор перемикавання (вибору) switch. Хід роботи програми може бути визначений значенням деякої змінної (перемикача). Такий альтернативний спосіб розгалуження програми заснований на використанні оператора перемикавання **switch**. Оператор **switch** містить блоки, що починаються зі слова **case**, після кожного **case** записано через пробіл те значення перемикача, за якого виконуваний цей блок. Останній блок починається зі слова **otherwise**, його оператори працюють у тому випадку, коли жоден із блоків **case** не був виконаний. Якщо хоча б один із блоків **case** виконаний, то відбувається вихід з оператора **switch** і перехід до оператора, наступного за **end**.

Припустимо, що потрібно знайти кількість одиниць і мінус одиниць у заданому масиві **i**, крім того, знайти суму всіх елементів, відмінних від одиниці і мінус одиниці. Лістинг програми містить файл-функцію, яка в заданому масиві повертає кількість мінус одиниць у першому вихідному аргументі, кількість одиниць – у другому, а суму – у третьому (рис. 2.47).

Для зупинки програми використовують оператор **pause** у таких формах:

- а) **pause** - зупиняє обчислення до натискання будь-якої клавіші;
- б) **pause(N)** - зупиняє обчислення на **N** секунд;
- в) **pause on** - вмикає режим паузи;
- г) **pause off** - вимикає режим паузи.

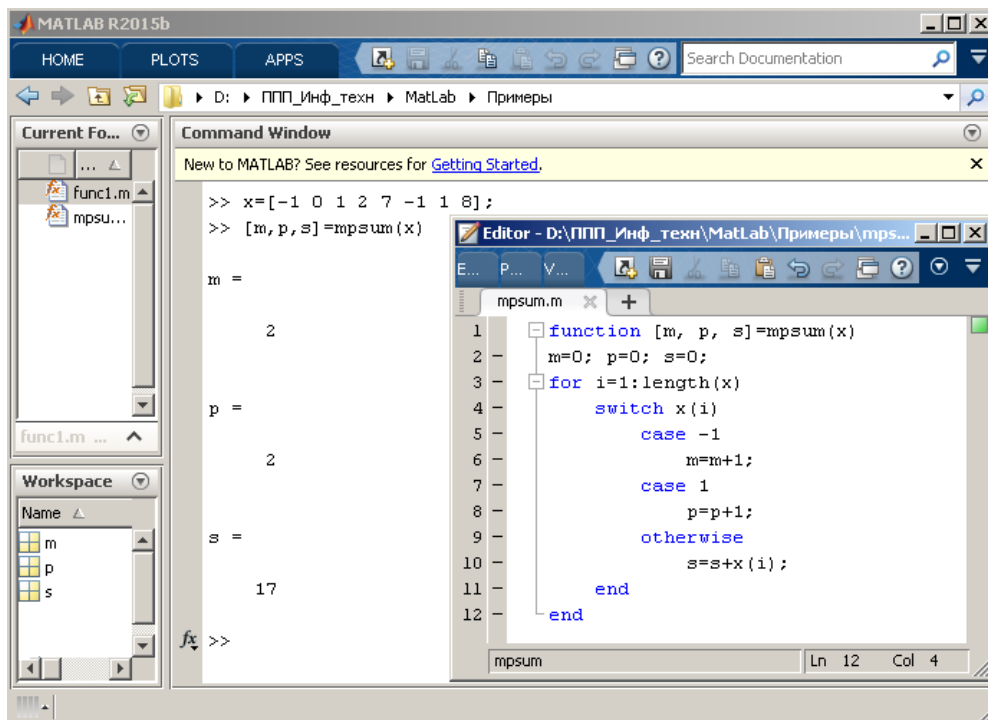


Рис. 2.47. Лістинг програми

2.5.4.6. Оператор переривання циклу *break*. Для організації циклічних обчислень потрібно дбати про те, щоб усередині циклу не виникло помилок. Наприклад, нехай задано масив x , що складається з цілих чисел, і потрібно сформувати новий масив y за правилом $y(i) = \frac{x(i+1)}{x(i)}$. Очевидно, що завдання може бути розв'язане за допомогою циклу **for**. Але якщо один з елементів вихідного масиву дорівнює нулю, то з розподілом вийде **inf**, і наступні обчислення можуть виявитися марними. Запобігти цій ситуації можна виходом із циклу, якщо поточне значення $x(i)$ дорівнює нулю. Наступний фрагмент програми демонструє використання оператора **break** для переривання циклу:

```

for x = 1:20
    z = x-8;
    if z==0
        break
    end
    y = x/z
end.
  
```

Як тільки змінна *z* набуває значення 0, цикл переривається. Оператор ***break*** дозволяє достроково перервати виконання циклів ***for*** і ***while***. Поза цими циклами оператор ***break*** не працює.

Якщо оператор ***break*** застосовано у вкладеному циклі, то він здійснює вихід тільки з внутрішнього циклу.

ЗАВДАННЯ ДЛЯ ВИКОНАННЯ

Завдання 1. Розробити файл-сценарій, у якому за допомогою функції діалогового введення ввести з клавіатури всі необхідні дані. Виконати розрахунок із використанням умовних операторів і вивести результати в командне вікно (табл. 2.30).

Таблиця 2.30

Варіанти завдання 1

Варіант	Завдання
1	Знайти суму додатних із чотирьох заданих змінних
2	Знайти максимальне значення з чотирьох заданих змінних і вивести її
3	Задано чотири змінні. Мінімальну замінити на суму інших
4	Задано чотири змінні. Підрахувати кількість від'ємних і кількість нульових із них
5	Знайти добуток від'ємних із чотирьох заданих змінних
6	Задано дві фігури: квадрат заданий довжиною сторони, а коло – довжиною радіуса. Визначити, яка з них має більшу площу і в скільки раз
7	Задано чотири змінні. Усі від'ємні з них замінити абсолютними значеннями і збільшити у два рази
8	Задано чотири змінні, підрахувати кількість тих, що дорівнюють нулю, додатних і від'ємних
9	Задано чотири змінні. Підрахувати кількість і добуток чисел, що потрапили в інтервал [1..5]
10	Задано чотири змінні. Усі додатні з них замінити від'ємними значеннями, помноженими на 5

Завдання 2. Написати файл-функцію з використанням операторів розгалуження і циклів, на підставі варіантів завдання, поданих у табл. 2.31. Вхідний масив A (вектор A) сформувати в командному вікні і передати функції як параметр. Матриці A і B (вектори A і B) виводити на екран в програмі.

Таблиця 2.31

Варіанти завдання 2

Варі- ант	Масив		Задача
	Вхідний	Сформувати	
1	2	3	4
1	$A_{3 \times 3}$	$B_{3 \times 3}, b_{ij} = \begin{cases} a_{ij}, i < j \\ a_{ji}^2, i \geq j \end{cases}$	Сформувати масив AI із мінімальних елементів рядків матриці A і масив BI із мінімальних елементів рядків матриці B . Серед елементів AI і BI знайти максимальний
2	A_3	$B_3, b_i = \sin(i^2), i = 1 \dots 3$	Сформувати масив C – суму елементів масивів A і B . Знайти максимальне значення масивів A, B, C
3	$A_{3 \times 3}$	$B_{3 \times 3}, b_{ij} = \sin(i) * \sin(j)$ $i = 1 \dots 3, j = 1 \dots 3.$	Визначити мінімальні елементи в матрицях A і B (mA і mB). Обчислити $C = A * B * mA * mB$
4	$A_{3 \times 3}$	$B_{3 \times 3}, b_{ij} = \begin{cases} 2*i+3*j, i=j, \\ 5*i+2*j, \text{иначе.} \end{cases}$	Сформувати масив AI із максимальних елементів рядків матриці A і масив BI із максимальних елементів рядків матриці B . Впорядкувати масив AI за зростанням, а масив BI – по спадній
5	$A_{3 \times 3}$	$B_3, b_i = \sin(i) + \cos(i)$ $i = 1 \dots 3.$	Визначити максимальні елементи в матриці A і масиві B (mA і mB). Обчислити $C = A*B*mA*mB$
6	$A_{3 \times 3}$	$B_3, b_i = \log(2i + \cos(i)),$ $i = 1 \dots 3.$	Сформувати масив AI із середніх значень елементів рядків матриці A .

1	2	3	4
			Впорядкувати масив A за зростанням, а B – по спадній. Здійснити поелементне множення A і B
7	$A_{3 \times 3}$	$B_3, b_i = \sin(\ln(i) + \cos(i)), i = 1 \dots 3.$	Замінити рядок матриці A , у якому знаходиться максимальний елемент, масивом B . Обчислити суму елементів отриманої матриці
8	A_3	$B_{3 \times 3}, b_{ij} = \begin{cases} 1 + \cos(i - j), i > j, \\ 1 - \sin(i + j), \text{иначе.} \end{cases}$	Масив A впорядкувати за зростанням і замінити ним останній рядок матриці B
9	A_3	$B_3, b_i = i * \log(i^2) + \sin(i), i = 1 \dots 3.$	Впорядкувати за зростанням масиви A і B . Здійснити поелементне ділення упорядкованих масивів. Визначити добуток елементів результуючого масиву
10	$A_{3 \times 3}$	$B_3, b_i = i * \sin(j) * \log(i) i = 1 \dots 3$	Обчислити добуток елементів матриці A (pA) і суму елементів матриці B (cB). Обчислити матрицю $C = pA * cB * A * B'$

Завдання 3. Написати програму в MATLAB для визначення свого віку на поточну дату. ПІБ, дату народження та поточну дату вводити з клавіатури.

Завдання 4. Розробити програму привітання Welcome, яка буде вас вітати залежно від часу доби, кажучи:

- «Доброго ранку!» між п'ятою і дванадцятою годинами;
- «Доброго дня» між 12 і 17 і т. д.,

і додасть «Я порадив (ла) б Вам спати в цей час!» після опівночі.

2.6. Використання MATLAB для розв'язання задач обчислювальної математики. Розв'язання типових задач алгебри та аналізу

Ознайомлення з можливостями системи MATLAB щодо розв'язання типових задач алгебри і аналізу та методикою їх розв'язання.

2.6.1. Розв'язання систем лінійних рівнянь

Нехай задана система n лінійних алгебраїчних рівнянь із n невідомими

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1 \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = b_2 \\ \dots + \dots + \dots + \dots = \dots \\ a_{n1}x_1 + a_{n2}x_2 + \dots + a_{nn}x_n = b_n \end{cases} \quad (2.1)$$

Ця система рівнянь у матричній формі подана так:

$$A \cdot X = B, \quad (2.2)$$

де A – квадратна матриця коефіцієнтів, розміром $n \times n$ рядків і стовпців;

X – вектор-стовпець невідомих ($n \times 1$);

B – вектор-стовпець правих частин ($n \times 1$).

Систему рівнянь (2.2) можна розв'язати різними методами. Один із найбільш простих і ефективних методів є метод виключення Гауса і його модифікації. Алгоритм методу заснований на приведенні матриці A до трикутного вигляду (*прямий хід*) і послідовному обчисленні невідомих (*зворотний хід*). Ці процедури можна виконувати над невираженими матрицями, в іншому випадку метод Гауса непридатний.

У системі MATLAB для розв'язання систем лінійних рівнянь передбачені операції

/	праве ділення
\	ліве ділення
⁻¹	піднесення до степеня -1
inv(A)	обернення матриці A

Вирази:

$$X' = B' / A',$$

$$X = A \backslash B,$$

$$X' = B' * A'^{-1},$$

$$X' = B' * \text{inv}(A')$$

дають розв'язок ряду систем лінійних рівнянь $A \cdot X = B$ ($A \cdot X' = B$), де A – матриця розміром $n \times n$, B - матриця (вектор) розміром $n \times 1$.

Якщо вектор B подано як рядок, його необхідно транспонувати.

Наприклад, розв'язати систему чотирьох лінійних рівнянь

$$\begin{cases} 1.1161x_1 + 0.1397x_2 + 0.1254x_3 + 0.1490x_4 = 1.5471; \\ 0.1582x_1 + 0.1768x_2 + 1.1675x_3 + 0.1871x_4 = 1.6471; \\ 0.1968x_1 + 1.2168x_2 + 0.2071x_3 + 0.2271x_4 = 1.7471; \\ 0.2368x_1 + 0.2568x_2 + 0.2471x_3 + 1.2671x_4 = 1.8471. \end{cases}$$

Щоб розв'язати систему лінійних рівнянь вигляду $A \cdot x = B$, де A – задана квадратна матриця розміром $n \times n$; B – заданий вектор стовпець розміром n , достатньо застосувати операцію «\» і обчислити вираз $A \backslash B$.

Операцію «\» називають лівим діленням матриць і із застосуванням до матриць A і B у вигляді $A \backslash B$ приблизно еквівалентна обчисленню виразу $\text{inv}(A) * B$. Тут як $\text{inv}(A)$ розуміють обчислення матриці, оберненої до матриці A . Операцію «/» називають правим діленням матриці. Вираз A / B приблизно відповідає обчисленню виразу $B * \text{inv}(A)$.

Розв'язання:

```
>> A=[1.1161 0.1397 0.1254 0.1490;
      0.1582 0.1768 1.1675 0.1871;
      0.1968 1.2168 0.2071 0.2271;
      0.2368 0.2568 0.2471 1.2671];
>> B=[1.5471; 1.6471; 1.7471; 1.8471];
>> X=A\B
X =
    1.0406
    0.9351
    0.9870
    0.8813
```

Після розв'язання обов'язково необхідно робити перевірку:

```
>> A*X
ans =
    1.5471
    1.6471
    1.7471
    1.8471
```

Розв'язують рівняння $F(x)=0$, або знаходять нулі функції, за допомогою функції **fzero (name, x₀)**. Як перший аргумент їй передано ім'я функції, яка задає вихідне рівняння, другим аргументом є початкове наближення до кореня.

Наприклад, визначимо нулі функції **cos(x)** на відрізку від 0 до π (рис. 2.48). Як початкове наближення приймемо $x_0 = 1$.

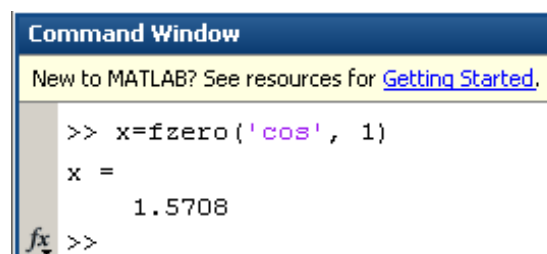


Рис. 2.48. Обчислення нулів функції **cos(x)**

Якщо потрібно знайти корінь функції, відмінної від стандартної, то спочатку створюють m-файл і потім користуються функцією **fzero**.

Наприклад, знайти корені рівняння $\cos(x) = x$ (рис. 2.49).

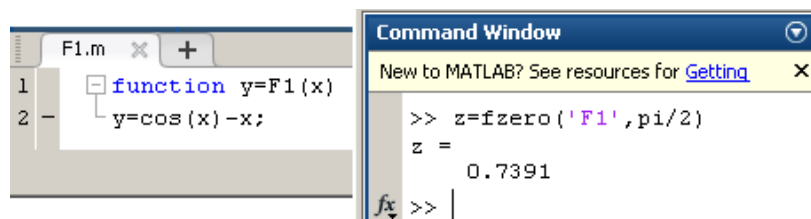


Рис. 2.49. Обчислення коренів рівняння $\cos(x) = x$

Якщо знайдено абсолютно точне значення кореня, то значення функції в цій точці дорівнює нулю. Отже, величина функції в наближено знайденому нулі опосередковано характеризує похибку результату. Щоб управляти похибкою, потрібно викликати функцію *fzero* з трьома аргументами **fzero(name,x₀,tol)**, де tol задає необхідну величину похибки (помилки). Часто буває необхідно знайти корінь функцій многочленів (у тому числі і комплексні). Для цього існує спеціальна функція **roots(p)**, де *p* – масив коефіцієнтів многочлена.

Наприклад, для многочлена $p = x^4 - 3x^3 + 3x^2 - 3x + 2$ потрібно спочатку сформувати масив його коефіцієнтів (за зменшенням ступеня *x*) (рис. 2.50).

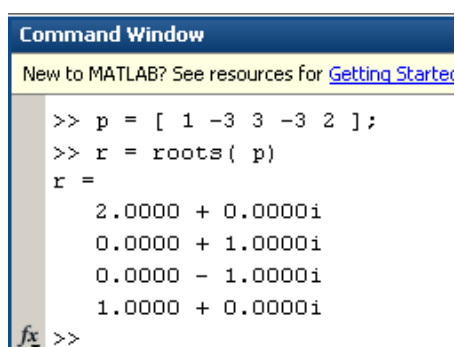


Рис. 2.50. Використання функції **roots(p)**

У задачі про знаходження нулів функції складним моментом є знаходження початкового наближення до нуля функції, а також завжди

априорне оцінювання їхньої кількості. Тому важливо паралельно із застосуванням функцій типу **roots** або **fzero** візуалізувати поведінку шуканих функцій на тому чи іншому відрізку значень аргументу.

2.6.2. Знаходження мінімуму заданих функцій

У системі MATLAB є спеціальні функції для пошуку мінімуму заданих функцій:

- **fminbnd** - пошук функції однієї змінної для фіксованого інтервалу;
- **fmincon** - пошук мінімуму нелінійної задачі з обмеженнями;
- **fminsearch** - пошук мінімуму функції кількох змінних без обмежень;
- **fminunc** - пошук мінімуму функції кількох змінних без обмежень.

При цьому можливий пошук мінімуму як для функції однієї дійсної змінної, так і для функцій багатьох змінних.

Для функцій однієї змінної їхні мінімуми знаходить функція $xmin = \text{fminbnd}(\text{name}, x_0, x_1)$. Тут *name* задає ім'я функції, у якій знаходяться мінімуми, а x_0 і x_1 задають відрізок пошуку. Для пошуку мінімуму функції кількох змінних застосовують функцію **fmins**: $xmins = \text{fminsearch}(\text{name}, x_0)$. Тут *name* є ім'ям функції декількох змінних, для якої шукають мінімум, а x_0 – це вектор її аргументів, із якого починається пошук.

Наприклад, створимо просту функцію двох змінних (запишемо її у вигляді файл-функції), яка має за мінімум точку (0,0). М-файлу присвоєне ім'я **MFunc1** (рис. 2.51).

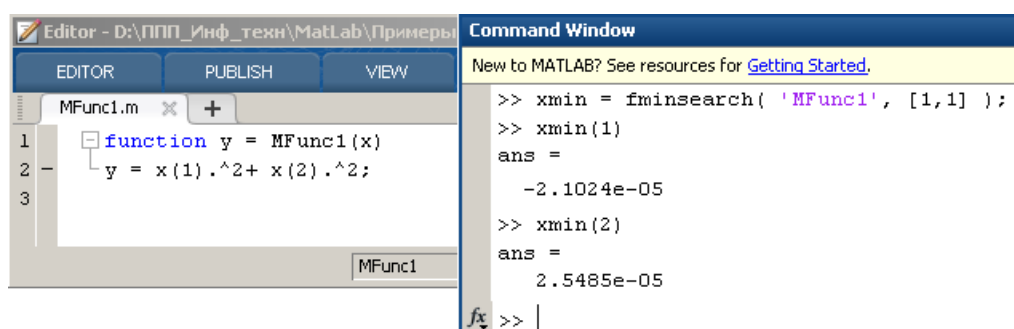


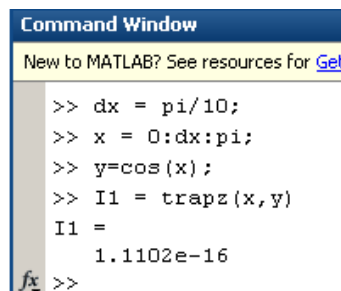
Рис. 2.51. Пошук мінімуму функції двох змінних

Після цього можна викликати функцію **fminsearch**, яка наближено знаходить вектор *xmin* координат точки мінімуму.

2.6.3. Обчислення інтегралів

Для обчислення інтегралів методом трапецій у системі MATLAB передбачена функція **trapz**: $Integ = trapz(x, y)$. Точність обчислення інтеграла залежить від кроку інтеграції.

Наприклад, обчислити $\int_0^{\pi} \cos(x) dx$ методом трапецій із кроком $\pi/10$ (рис. 2.52).



```
Command Window
New to MATLAB? See resources for Get
>> dx = pi/10;
>> x = 0:dx:pi;
>> y=cos(x);
>> I1 = trapz(x,y)
I1 =
    1.1102e-16
fx >>
```

Рис. 2.52. Обчислення інтегралів методом трапецій

Зазвичай для досягнення високої точності потрібно виконувати інтегрування з дуже малими кроками, а контролювати досягнуту точність порівнянням послідовних результатів. За одного і того самого кроку інтегрування методи більш високих порядків точності досягають більш точних результатів. У системі MATLAB методи інтегрування більш високих порядків точності реалізовані функціями **quad(@fun,a,b,tol)** (метод Симпсона) і **quad8** (метод Ньютона-Котеса восьмого порядку точності).

Подвійні інтеграли в системі MATLAB обчислюють за допомогою спеціальної функції **dblquad**.

Наприклад, обчислити інтеграл $\int_0^1 \int_1^2 (x \sin(y) + y \sin(x)) dx dy$ (рис. 2.53).

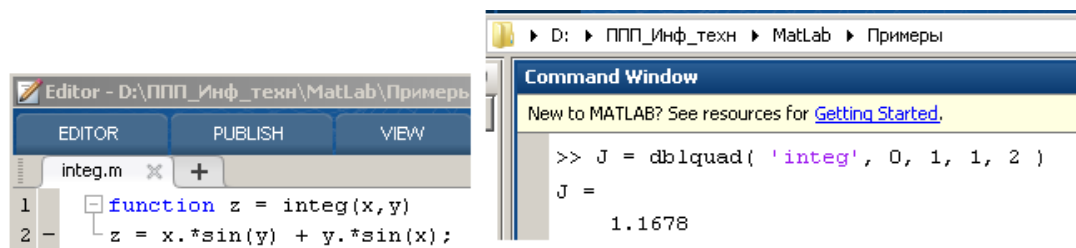


Рис. 2.53. Обчислення подвійного інтеграла за допомогою функції **dblquad**

2.6.4. Символьні обчислення і операції Symbolic Math Toolbox

Можливості вбудованого пакета символьних обчислень і операцій Symbolic Math Toolbox достатньо великі. Ми розглянемо лише деякі його можливості. Символьний об'єкт створюють за допомогою функції **syms**. Команда створює три символьні змінні **x**, **a** і **b**:

» **syms x a b** .

Конструювання символьних функцій від змінних класу **sym** object відбувається з використанням звичайних арифметичних операцій і позначень для вбудованих математичних функцій. Запис формули для виразу в один рядок не завжди зручний, більш природний вигляд виразу виводить у командне вікно функція **pretty** (рис. 2.54).

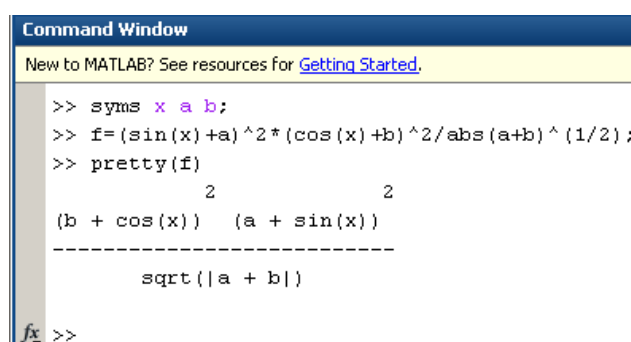
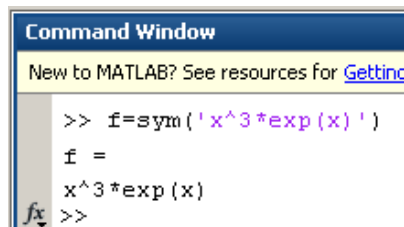


Рис. 2.54. Демонстрація роботи функції **pretty**

Символьну функцію можна створити без попереднього оголошення змінних за допомогою **sym**, вхідним аргументом якої є рядок із виразом, укладений в апострофи (рис. 2.55).



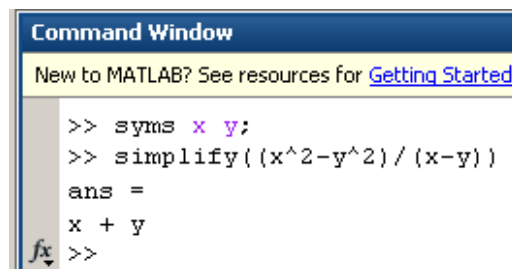
```
Command Window
New to MATLAB? See resources for Getting Started.

>> f=sym('x^3*exp(x)')
f =
x^3*exp(x)
fx >>
```

Рис. 2.55. Демонстрація роботи функції **sym**

2.6.5. Спрощення виразів

Спростимо вираз $\frac{x^2 - y^2}{x - y}$, використовуючи функцію **simplify** (рис. 2.56).

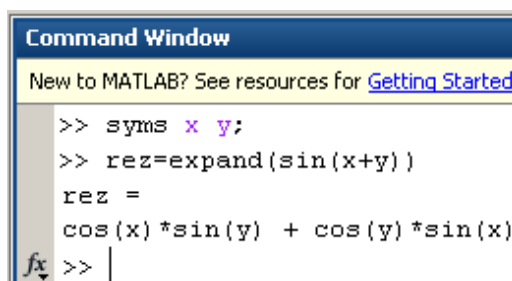


```
Command Window
New to MATLAB? See resources for Getting Started.

>> syms x y;
>> simplify((x^2-y^2)/(x-y))
ans =
x + y
fx >>
```

Рис. 2.56. Демонстрація роботи функції **simplify**

Спрощення тригонометричних, логарифмічних, експоненціальних функцій і поліномів здійснюється функцією **expand**: **rez = expand(S)**, де **S** - символічний вираз, який потрібно спростити; **rez** – результат (рис. 2.57).



```
Command Window
New to MATLAB? See resources for Getting Started.

>> syms x y;
>> rez=expand(sin(x+y))
rez =
cos(x)*sin(y) + cos(y)*sin(x)
fx >> |
```

Рис. 2.57. Демонстрація роботи функції **expand**

За допомогою функції **factor** можна розкласти многочлени на прості множники (рис. 2.58, а), а цілі числа - у добуток простих чисел (рис. 2.58, б).

```
Command Window
New to MATLAB? See resources for Getting Started.

>> factor(sym('x^5 - 1'))
ans =
[ x - 1, x^4 + x^3 + x^2 + x + 1]
fx >> |
```

а

```
Command Window
New to MATLAB? See resources for Getting Started.

>> factor(sym('36'))
ans =
[ 2, 2, 3, 3]
fx >> |
```

б

Рис. 2.58. Демонстрація роботи функції **factor**

Функція **subs** підставляє нові вирази для вказаних символічних змінних (рис. 2.59).

```
Command Window
New to MATLAB? See resources for Getting Started.

>> syms x y a b
>> subs(x*y, [x, y], [a+b, a-b])
ans =
(a + b) * (a - b)
fx >> |
```

Рис. 2.59. Демонстрація роботи функції **subs**

Symbolic Math Toolbox дає змогу працювати як із невизначеними інтегралами, так і визначеними. Невизначені інтеграли від символічних функцій обчислюють за допомогою функції **int**, як входні аргументи вказують символічну функцію і змінну, за якою відбувається інтегрування (рис. 2.60).

```
Command Window
New to MATLAB? See resources for Getting Started.

>> f=x^3*exp(x)-3*x^2*exp(x)+6*x*exp(x)-6*exp(x);
>> I=int(f,x)
I =
exp(x)*(x^3 - 6*x^2 + 18*x - 24)
>> pretty(I)
      3      2
exp(x) (x  - 6 x  + 18 x - 24)
fx >> |
```

Рис. 2.60. Демонстрація роботи функцій **int** і **pretty**

Ще декілька функцій, що часто використовують для символічних обчислень:

inv - обчислює обернену матрицю;

limit – обчислює граничні значення;

taylor - розкладає функції в ряд Тейлора;

solve - розв'язує рівняння алгебри і систему рівнянь алгебри [4-7].

Для роботи із символічними даними передбачена оболонка (додаток) **funtool**. Вона є інтерактивним графічним калькулятором, що допомагає швидко побудувати графіки двох функцій $f(x)$ і $g(x)$. Із запуском виведені три автономні вікна: два графічних і керуюче (рис. 2.61). Керуюче вікно містить два поля для введення функцій, поле введення граничного значення змінної x , поле введення масштабуючого коефіцієнта.

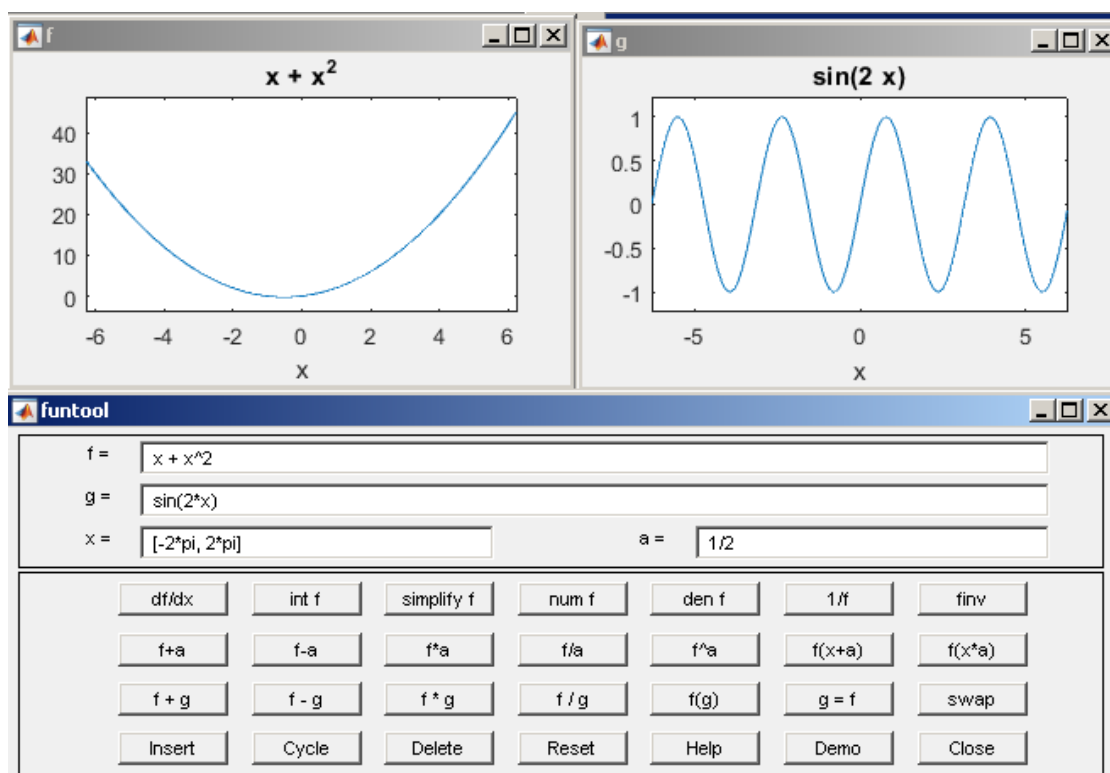


Рис. 2.61. Інтерфейс додатка **funtool**

ЗАВДАННЯ ДЛЯ ВИКОНАННЯ

Завдання 1. Скласти програми для знаходження коренів рівняння $f_1(x) = 0$ і $f_2(x) = 0$ і вивести графіки функції на підставі завдання з табл. 2.32. Перевірити отримані корені. Початкове наближення задати самому.

Таблиця 2.32

Варіанти завдання 1

Варіант	$f_1(x)$ – поліном третього степеня з коефіцієнтами a				$f_2(x)$
	2	3	4	5	6
1	0	-1	4	-1	$0.2e^x - 20$
2	0	2	-2	-15	$40 \cdot \cos(x)$
3	0	1	4	-1	$10 \cdot \log(x + 5.5)$
4	0	9	-8	-70	$100 \cdot \sin(x)$
5	0	-4	4	50	$70 \cdot \cos(x)$
6	1	-5	4	40	$60e^{0.1x} - 100$
7	2	-3	2	30	$20 \cdot \sin(2x)$
8	3	-6	1	50	$e^{ x } \sin(2x)$
9	4	-9	1	70	$e^{ x } \cos(3x)$
10	5	-7	5	60	$-60 \cdot \cos(x)$
11	-1	-4	9	60	$15 \cdot \log(x + 1.5)$
12	-2	-6	-7	55	$-50 \cdot \lg(x + 1.5)$
13	-3	-9	-8	75	$-100 \cdot \cos(x)$
14	-4	7	8	-75	$20 \cdot \sin(x/2)$
15	-5	1	4	-1	$40 \cdot \cos(x/2)$
16	-4	5	3	1	$\sin(1/x)x^4$
17	1	8	-8	-50	$x \cdot \sin(2x)$
18	6	-2	1	20	$21 \cdot \sin(x^2)$
19	1	4	9	21	$15x \cdot \ln(x + 0.3)$
20	2	6	7	22	$-x^2 \cdot \lg(x + 1.5)$

Завдання 2. Розв'язати систему лінійних алгебраїчних рівнянь за допомогою чотирьох операторів. Обчислити два варіанти, номер завдання взяти з табл. 2.33. Перевірити отримані корені.

Таблиця 2.33

Варіанти завдання 2

Варі- ант	Матриця	Век- тор	Варі- ант	Матриця	Век- тор
1	$a_{i,j}$ $= \begin{bmatrix} 2 & -4 & 3 & 1 \\ -1 & 5 & -7 & -3 \\ 10 & -2 & 4 & 4 \\ -1 & 1 & -1 & -1 \end{bmatrix}$	b_i $= \begin{bmatrix} 7 \\ 24 \\ 34 \\ -6 \end{bmatrix}$	2	$a_{i,j}$ $= \begin{bmatrix} 5 & -3 & 4 & -2 \\ 10 & 3 & -4 & 2 \\ 7 & -5 & 8 & -10 \\ 4 & 5 & -8 & 10 \end{bmatrix}$	b_i $= \begin{bmatrix} 4 \\ 11 \\ 0 \\ 11 \end{bmatrix}$
3	$a_{i,j}$ $= \begin{bmatrix} 2 & 2 & -1 & 1 \\ 4 & 3 & -1 & 2 \\ 8 & 5 & -3 & 4 \\ 3 & 3 & -2 & 2 \end{bmatrix}$	b_i $= \begin{bmatrix} 4 \\ 6 \\ 12 \\ 6 \end{bmatrix}$	4	$a_{i,j}$ $= \begin{bmatrix} 1 & 1 & -6 & 4 \\ 3 & -1 & -6 & -4 \\ 2 & 3 & 9 & 2 \\ 3 & 2 & 3 & 8 \end{bmatrix}$	b_i $= \begin{bmatrix} 6 \\ 2 \\ 6 \\ -7 \end{bmatrix}$
5	$a_{i,j}$ $= \begin{bmatrix} 1 & 5 & 3 & 4 \\ 7 & 14 & 20 & 27 \\ 5 & 10 & 16 & 19 \\ 3 & 5 & 6 & 13 \end{bmatrix}$	b_i $= \begin{bmatrix} 0 \\ 0 \\ -2 \\ 5 \end{bmatrix}$	6	$a_{i,j}$ $= \begin{bmatrix} 2 & 4 & 3 & 4 \\ 4 & -2 & 5 & 6 \\ 6 & -3 & 7 & 8 \\ 8 & -4 & 9 & 10 \end{bmatrix}$	b_i $= \begin{bmatrix} 5 \\ 7 \\ 9 \\ 11 \end{bmatrix}$
7	$a_{i,j}$ $= \begin{bmatrix} 1 & 1 & 1 & 1 \\ 2 & 3 & -1 & 2 \\ -1 & 2 & 4 & 1 \\ 2 & -1 & 1 & -1 \end{bmatrix}$	b_i $= \begin{bmatrix} 2 \\ 4 \\ 5 \\ 9 \end{bmatrix}$	8	$a_{i,j}$ $= \begin{bmatrix} 5 & -3 & 2 & 4 \\ 4 & -2 & 3 & 7 \\ 8 & -6 & -1 & -5 \\ 7 & -3 & 7 & 17 \end{bmatrix}$	b_i $= \begin{bmatrix} 7 \\ 10 \\ 4 \\ 17 \end{bmatrix}$
9	$a_{i,j}$ $= \begin{bmatrix} 1 & 2 & 15 & 11 \\ -2 & 4 & 3 & 61 \\ -3 & -8 & 11 & 12 \\ 15 & 7 & 8 & -4 \end{bmatrix}$	b_i $= \begin{bmatrix} 2 \\ 4 \\ 5 \\ 9 \end{bmatrix}$	10	$a_{i,j}$ $= \begin{bmatrix} 5 & 3 & -4 & -15 \\ 6 & 1 & 10 & 2 \\ 1 & 5 & 3 & 8 \\ -4 & 5 & -1 & 9 \end{bmatrix}$	b_i $= \begin{bmatrix} 6 \\ 9 \\ 12 \\ -9 \end{bmatrix}$

Завдання 3. Знайти визначений інтеграл для підінтегральної функції, заданої в табл. 2.34, методами трапеції та Сімпсона.

Таблиця 2.34

Варіанти завдання 3

Варі-ант	Функція	Інтер-вал	Варі-ант	Функція	Інтер-вал
1	$f(x) = x^2 - e^x$	[-2, 2]	2	$f(x) = x - \cos(x)$	[-2, 2]
3	$f(x) = \sin(x) - 2\cos(x)$	[-2, 2]	4	$f(x) = \cos(x) + (1 + x^2)^{-1}$	[-2, 2]
5	$f(x) = (x - 2)^2 - \ln(x)$	[0.5, 4.5]	6	$f(x) = 2x - \operatorname{tg}(x)$	[-1.4, 1.4]
7	$f(x) = x^2 - 20\cos(x + 1)$	[-5, 5]	8	$f(x) = 20\ln(x^2 + 1) - 0.1x^3$	[-5, 15]
9	$f(x) = x^2 - e^x$	[-4, 2]	10	$f(x) = 2x - \cos(x)$	[-2, 2]

Завдання 4. Обчислити значення визначеного інтегралу методами трапецій і Сімпсона. Дані взяти з табл. 2.35.

Таблиця 2.35

Варіанти завдання 4

Варіант	Підінтегральна функція	Інтервал інтегрування	Точність обчислень	Варіант	Функція	Інтервал інтегрування	Точність обчислень
1	$\ln x / x \sqrt{1 + \ln x}$	[1;3.5]	0.001	2	$\operatorname{tg}^2 x + c \operatorname{tg}^2 x$	[- $\pi/6$; $\pi/3$]	0.002
3	$1/x \ln x$			4	$\ln^2 x / x$	[1.0;4.0]	0.003
5	$\sqrt{e^x - 1}$	[0;ln2]	0.0015	6	$x e^x \sin x$	[1.0;4.0]	0.002
7	$x \frac{e^x - e^{-x}}{2}$	[0;2]	0.001	8	$1/\sqrt{9 + x^3}$	[2.0;5.0]	0.001
9	$\sin(1/x) x^4$	[1.0;2.5]	0.0005	10	$x^3 \operatorname{arctg} x$	[0; $\sqrt{3}$]	0.003

Завдання 5. Створити символний об'єкт і записати формулу в більш звичному вигляді (табл. 2.36).

Таблиця 2.36

Варіант 1	Варіант 2
$f = \frac{(\sin x + a)^2 \cos^2(x + b)}{ \sqrt{a + b} }$	$z = \frac{(d + \operatorname{tg} x)^2 \left(e^x + \frac{c}{d} \right)^3}{\left \sqrt[5]{c + \frac{d}{c}} \right }$

Завдання 6. Використовуючи оболонку **funtool** (графічний калькулятор), побудувати графіки двох функцій із табл. 2.34 за своїм і наступним варіантами.

2.7. Апроксимація та інтерполяція даних. Методи розв'язання звичайних диференціальних рівнянь

Вивчення основних методів апроксимації та інтерполяції даних у системі MATLAB і способів розв'язання звичайних диференціальних рівнянь.

2.7.1. Інтерполяція і апроксимація даних

2.7.1.1. Під апроксимацією зазвичай розуміють опис деякої, часом не заданої явно, залежності або сукупності, що подають її дані за допомогою іншої, зазвичай більш простої або більш однакової, залежності. Часто дані знаходяться у вигляді окремих вузлових точок, координати яких задані таблицею даних [4-7].

Результат апроксимації може не проходити через вузлові точки. Навпаки, завдання інтерполяції - знайти дані в найближчому колі (околі) вузлових точок. Для цього використовують відповідні функції, значення яких у вузлових точках збігаються з координатами цих точок. Наприклад, за *лінійної інтерполяції* залежності $y(x)$ вузлові точки з'єднуються одна з одною відрізками прямих, і вважають, що шукані проміжні точки розташовані на цих відрізках.

Для підвищення точності інтерполяції застосовують параболи (*квадратична інтерполяція*) або поліноми більш високого ступеня (*поліноміальна інтерполяція*). Для обробки даних MATLAB використовує різні функції інтерполяції та апроксимації даних [4-7].

Поліноміальна апроксимація

Одна з найбільш відомих апроксимацій – поліноміальна. У системі MATLAB визначені функції апроксимації даних поліномами за методом найменших квадратів – поліноміальної регресії. Це виконує функція:

• $p = \text{polyfit}(x,y,n)$ – знаходить коефіцієнти полінома $p(x)$ ступеня n , який апроксимує функцію $y(x)$ у сенсі методу найменших квадратів. Результатом є рядок p довжини $n + 1$, яка містить коефіцієнти полінома, що апроксимує цю функцію;

• $[p,S] = \text{polyflt}(x,y,n)$ – повертає коефіцієнти полінома p і структуру S для використання разом із функцією **polyval** для оцінювання або передбачення похибки;

• $[p,S] = \text{polyflt}(x,y,n,mu)$ повертає коефіцієнти полінома p і структуру S для використання разом із функцією **polyval** для оцінювання або передбачення похибки, але так, що відбувається центрування (нормування) і масштабування x , $xnorm = (x - mu(1))/mu(2)$, де $mu(1) = \text{mean}(x)$ і $mu(2) = \text{std}(x)$. Центрування і масштабування не тільки покращують властивості степеневого многочлена, одержаного за допомогою **polyval**, але і значно підвищують якісні характеристики самого алгоритму апроксимації.

Графік полінома третього ступеня, що апроксимує функцію, на рис. 2.62 показаний суцільною лінією, а точки вихідної залежності позначені кружками. На жаль, за ступеня полінома більше 5 похибка поліноміальної регресії (і апроксимації) сильно зростає, і її застосування без центрування і масштабування стає ризикованим. Зверніть увагу на те, що за поліноміальної регресії вузлові точки не лягають точно на графік полінома, оскільки їх наближення до нього є найкращим у сенсі мінімального середньоквадратичного відхилення.

2.7.1.2. Інтерполяція періодичних функцій рядом Фур'є. Під інтерполяцією зазвичай розуміють обчислення значень функції $f(x)$ у проміжках між вузловими точками. Лінійна, квадратична і поліноміальна

інтерполяція будуть реалізовані за поліноміальної апроксимації. А ось для періодичних (і особливо гладких періодичних) функцій хороші результати може дати їх інтерполяція тригонометричним рядом Фур'є. Для цього використовують таку функцію: $y = \text{interpft}(x,n)$ - повертає вектор y , що містить значення періодичної функції, визначені в n рівномірно розташованих точках (рис. 2.63). Якщо $\text{length}(x) = rr$; і x має інтервал дискретизації dx , то інтервал дискретизації для y складає $dy = dx*m/n$, причому n не може бути менше, ніж r . Якщо X - матриця, interpft оперує стовпцями X , повертаючи матрицю Y із такою самою кількістю стовпців, як і в X , але з n рядками. Функція $y = \text{interpft}(x,n,\text{dim})$ працює або з рядками, або за допомогою стовпців залежно від значення параметра dim [4-7].

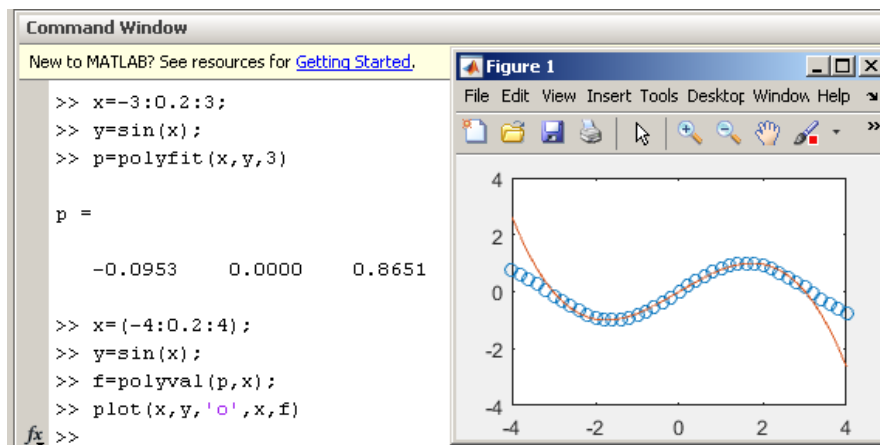


Рис. 2.62. Приклад використання функції *polyfit*

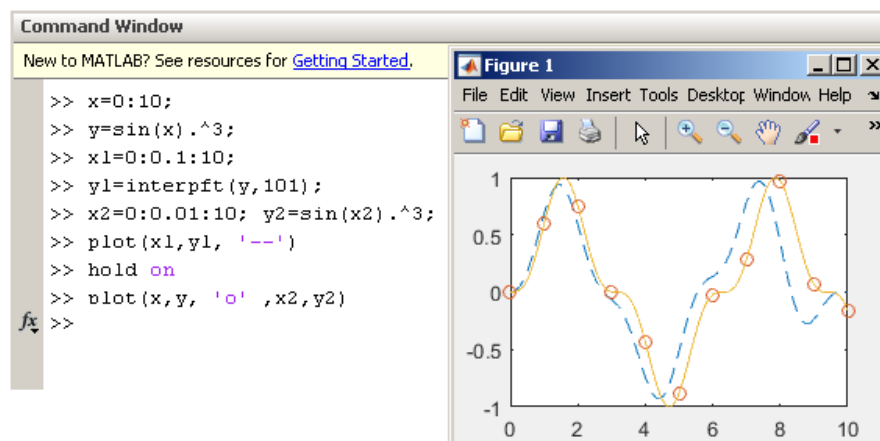


Рис. 2.63. Приклад використання функції *interpft*

2.7.1.3. Одновимірна таблична інтерполяція. У ряді випадків дуже зручна сплайнова інтерполяція і апроксимація таблично заданих функцій, за якої проміжні точки шукають за відрізками поліномів третього ступеня – це кубічна сплайнова інтерполяція. При цьому зазвичай такі поліноми обчислюють так, щоб не тільки їхні значення збігалися з координатами вузлових точок, але щоб у вузлових точках були безперервні похідні першого і другого порядків. Така поведінка характерна для гнучкої лінійки, закріпленої у вузлових точках, звідки і походить назва *spline* (сплайн) для цього виду інтерполяції (апроксимації). Для одновимірної табличної інтерполяції використовують функцію **interp1**:

• $y_i = \text{interp1}(x, y, x_i)$ – повертає вектор y_i , що містить елементи, відповідні елементам x_i і отримані інтерполяцією векторів x і y (рис. 2.64). Вектор x визначає точки, у яких задано значення y . Якщо Y - матриця, то інтерполяція виконується для кожного стовпця Y , і y_i має довжину $\text{length}(x_i)$;

• $y_i = \text{interp1}(x, y, x_i, \text{метод})$ – дає змогу за допомогою параметра *method* задати метод інтерполяції:

- 'nearest' – інтерполяція поліномами нульового ступеня;
- 'linear' – інтерполяція поліномами першого ступеня (за замовчуванням);
- 'cubic' – інтерполяція кубічним поліномом;
- 'spline' – інтерполяція кубічним сплайном;
- 'pchip' – інтерполяція кубічним ермітовим сплайном.

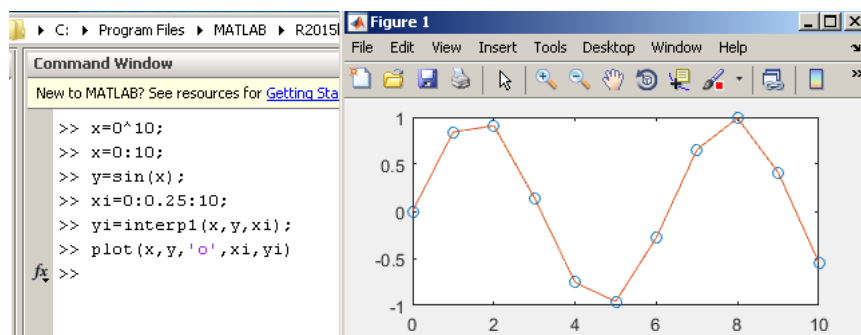


Рис. 2.64. Приклад візуалізації процесу інтерполяції функції *sin*

2.7.1.4. Апроксимація рядом методів точок функції. У MATLAB передбачена апроксимація рядом методів точок функції, графік якої побудований. Для цього в позиції Tools графічного вікна є дві нові команди:

Basic Fitting – основні види апроксимації (регресії);

Data Statistics – статистичні параметри даних.

Команда **Basic Fitting** (рис. 2.65) відкриває вікно, що дає доступ до ряду видів апроксимації і регресії: сплайнової, ермітової і поліноміальної зі ступенями від 1 (лінійна апроксимація) до 10, у тому числі зі ступенем 2 (квадратична апроксимація) і 3 (кубічна апроксимація).

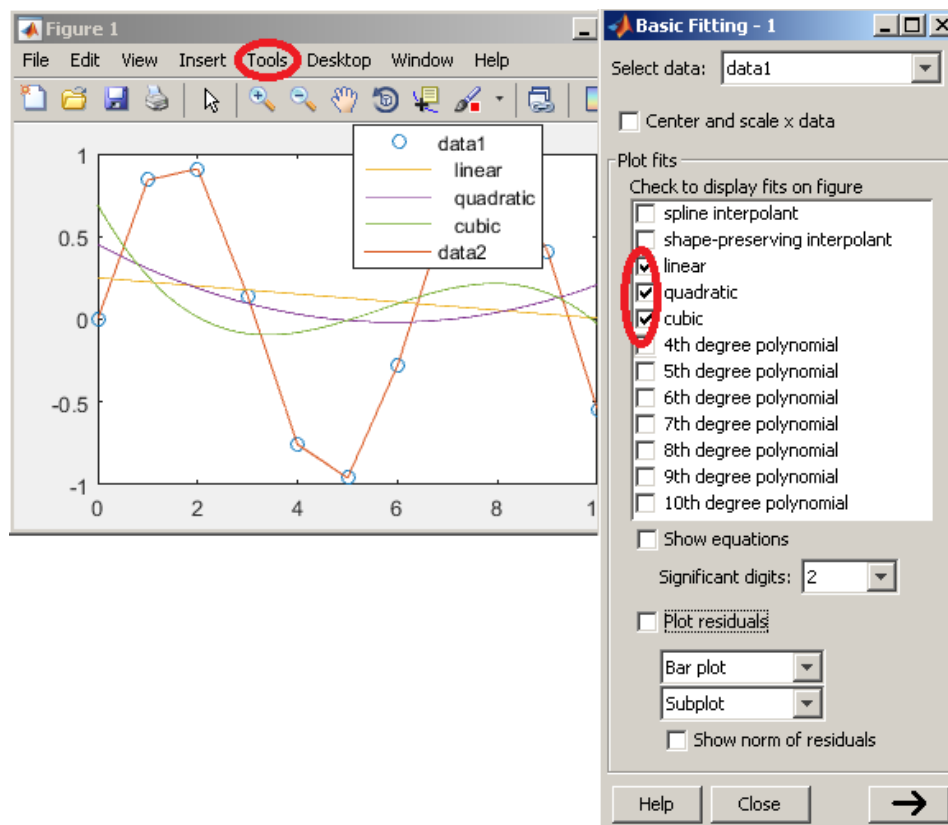


Рис. 2.65. **Basic Fitting** – основні види апроксимації (регресії)

Команда **Data Statistics** відкриває вікно з результатами найпростішої статистичної обробки даних (рис. 2.66) [4-7].

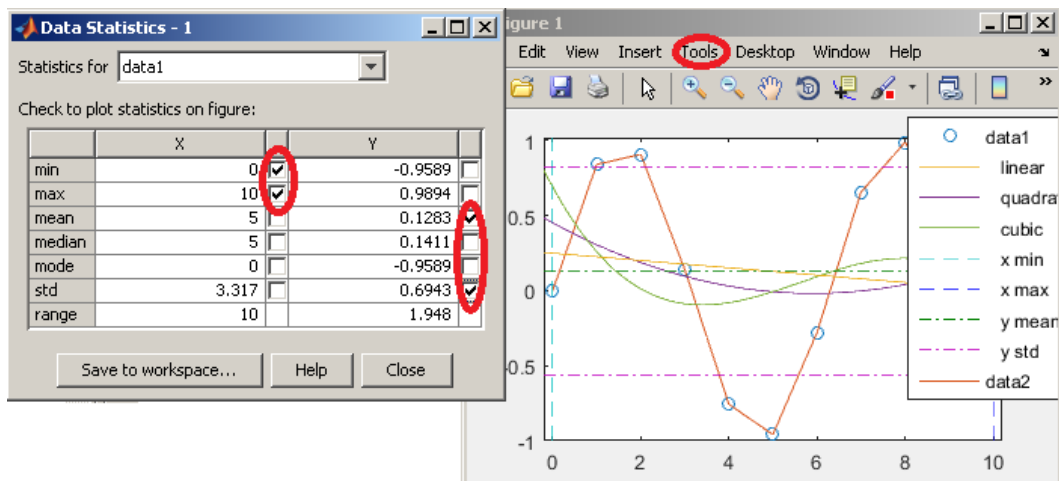


Рис. 2.66. **Data Statistics** – результати найпростішої статистичної обробки даних

2.7.2. Розв'язання звичайних диференціальних рівнянь (ЗДР)

Для чисельного розв'язання звичайних диференціальних рівнянь (ЗДР) і систем довільного порядку із заданими початковими умовами в MATLAB використовують спеціальні вбудовані функції, які називають солверами. Система має досить великий набір солверів, які ґрунтовані на використанні різних числових методів [4-7].

Аналіз поведінки багатьох систем і пристроїв у динаміці базований на розв'язанні систем звичайних диференціальних рівнянь. Їх зазвичай подають як систему диференціальних рівнянь першого порядку у формі Коші:

$$\begin{aligned} \frac{dy}{dt} &= y'; \\ y' &= f(x, y), \quad y(t_0, t_{end}) = b. \end{aligned} \quad (2.3)$$

де t_0, t_{end} – початкові і кінцеві точки інтервалів.

Параметр t не обов'язково означає час, хоча найчастіше розв'язок диференціальних рівнянь шукають у часовій області.

Вектор b задає початкові і кінцеві умови.

Задача Коші для диференціального рівняння полягає у знаходженні функції, яка задовольняє диференціальне рівняння довільного порядку $y^{(n)} = f(t, y, y', \dots, y^{(n-1)})$ і початкові умови за $t = t_0$ $y(t_0) = u_0$, $y'(t_0) = u_1, \dots$, $y^{(n-1)}(t_0) = u_{n-1}$.

Порядок розв'язання задач Коші в MATLAB такий:

- 1) зведення диференціального рівняння до системи диференціальних рівнянь першого порядку;
- 2) написання файл-функції для системи рівнянь;
- 3) використання необхідного солвера;
- 4) візуалізація результату.

2.7.2.1. Функції для розв'язання диференціальних рівнянь і систем. Для розв'язання диференціальних рівнянь і систем у MATLAB передбачені такі солвери (функції):

ode45 (f, interval, X0, [options])	ode23 (f, interval, X0, [options])
ode113 (f, interval, X0, [options])	ode15s (f, interval, X0, [options])
ode23s (f, interval, X0, [options])	ode23t (f, interval, X0, [options])
ode23tb (f, interval, X0, [options]).	

Вхідними параметрами цих функцій є:

- f – вектор-функція для обчислення правої частини рівняння системи рівнянь;
- **interval** – масив із двох чисел, що визначає інтервал інтегрування диференціального рівняння або системи;
- $X0$ – вектор початкових умов системи диференціальних рівнянь;
- **options** – параметри управління ходом розв'язання диференціального рівняння або системи.

Усі функції повертають:

- масив T – координати вузлів сітки, де шукають розв'язок;
- матрицю X , i -й стовпець якої є значенням вектор-функції розв'язання у вузлі T_i .

Функцією **ode45** реалізований метод Рунге-Кутта четвертого-п'ятого порядку точності, функцією **ode23** також реалізований метод Рунге-Кутта, але другого-третього порядку, а функцією **ode113** реалізовано метод Адамса.

Для розв'язання жорстких систем призначені функція **ode15s**, у якій реалізований метод Гіра, і функція **ode23s**, що реалізує метод Розенброка. Для отримання більш точного розв'язку жорсткої системи краще використовувати функцію **ode15s**.

Для розв'язання системи з невеликою кількістю жорсткості можна використовувати функцію **ode23t**, а для грубого оцінювання подібних систем – функцію **ode23tb**.

2.7.2.2. Методика розв'язання диференціального рівняння другого порядку із застосуванням солвера ode45. Методику розв'язання диференціального рівняння другого порядку із застосуванням солвера **ode45** розглянемо на прикладі.

Приклад 1. Розв'язати задачу Коші $\frac{dy}{dx} = 2(x^2 + y^2)$, $y(0)=1$, $0 < x < 0,4$ і побудувати графік.

Розв'яжемо задачу за допомогою програми **ode45**. Спочатку запишемо праву частину рівняння в М-файл, сам М-файл розроблений як файл-функція з іменем **RZK**:

```
function f=RZK(x,y)
f = 2*(x^2+y^2).
```

Для чисельного розв'язання задачі Коші у вікні команд набирають такі оператори (команди):

```
>>[X Y] = ode45(@F, [0 0.4], [1]);
>>plot (X,Y).
```

Рівняння розв'язане, графік побудований (рис. 2.67).

Символьне розв'язання звичайних диференціальних рівнянь довільного порядку здійснює функція **dsolve**
 $r = \text{dsolve}('eq1,eq2,...','cond1,cond2,...','v').$

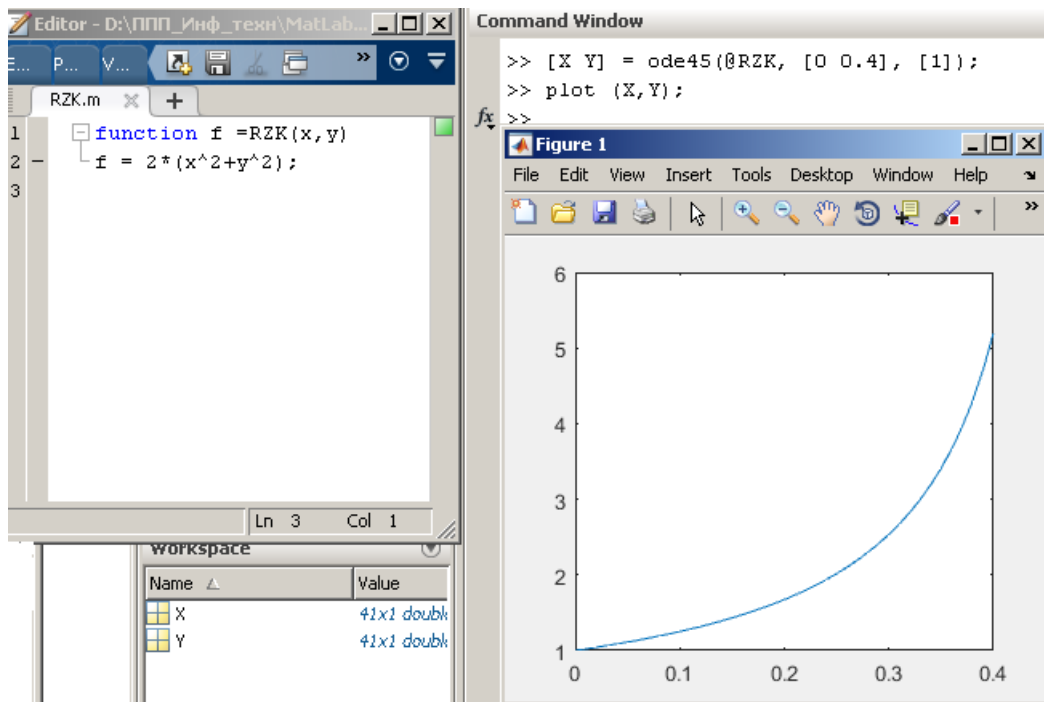


Рис. 2.67. Результати розв'язання задачі

2.7.2.3. Аналітичний вираз розв'язання ЗДР. Для розв'язання диференціальних рівнянь у символьному вигляді в MATLAB використовують функцію ***dsolve***, яка має такі формати звернення:

1. $y = \text{dsolve}('Dy(x)')$, де $Dy(x)$ — рівняння; y — рішення, що повертає функція ***dsolve***.

2. $y = \text{dsolve}('Dy(x)', 'ПУ')$, де $Dy(x)$ — рівняння; ПУ — початкові умови. Використовуючи функцію, потрібно дотримуватися певних обмежень щодо форми запису рівняння (або системи рівнянь). Так, якщо невідома функція позначена символічною змінною y , то її похідні позначають як $D[n]y$, де у квадратних дужках вказано порядок похідної. Отже, похідна має бути позначена в символьному вираженні Dy , друга похідна — $D2y$ і т. д. Функцію ***dsolve*** можна викликати з різним набором параметрів залежно від типу задачі, що розв'язують, і порядку системи рівнянь.

Загальний розв'язок рівняння $y'(x) + 0.33y(x) = e^{-x}$ матиме вигляд, зображений на рис. 2.68,

```

Command Window
New to MATLAB? See resources for Getting Started.

>> syms y;
>> S=dsolve('Dy=-0.33*y+exp(-x)', 'x')

S =

C7*exp(-(33*x)/100) - (100*exp(-(33*x)/100)*exp(-(67*x)/100))/67

fx >>

```

Рис. 2.68. Використання функції **dsolve**

або, використавши функцію **pretty** для подання символічних виразів у легшому для сприйняття виді, на рис. 2.69.

```

Command Window
New to MATLAB? See resources for Getting Started.

>> pretty(S)

          / 33 x \      / 67 x \
        exp| - ---- | exp| - ---- | 100
          \ 100 /      \ 100 /
C7 exp| - ---- | - -----
          \ 100 /              67

fx >>

```

Рис. 2.69. Використання функції **pretty**

Із початковими умовами $y(0) = 0$ розв'язок буде знайдено як такий, що зображений на рис. 2.70.

```

Command Window
New to MATLAB? See resources for Getting Started.

>> syms y;
>> S=dsolve('Dy=-0.33*y+exp(-x)', 'y(0)=0', 'x')

S =

(100*exp(-(33*x)/100))/67 - (100*exp(-(33*x)/100)*exp(-(67*x)/100))/67

>> pretty(S)

          / 33 x \      / 33 x \      / 67 x \
        exp| - ---- | 100  exp| - ---- | exp| - ---- | 100
          \ 100 /      \ 100 /      \ 100 /
-----
          67              67

fx >>

```

Рис. 2.70. Загальний розв'язок рівняння

2.7.3. Розв'язання систем диференціальних рівнянь (СДР)

2.7.3.1. Функції для розв'язання систем диференціальних рівнянь.

Перейдемо до опису функцій для розв'язання систем диференціальних рівнянь:

- $[T, Y] = \text{solver} (@F, \text{tspan}, y0)$ – де замість *solver* підставляємо ім'я конкретного m-файлу, де знаходиться розв'язок – інтегрує систему диференціальних рівнянь вигляду $y' = F(t, y)$ на інтервалі *tspan* із початковими умовами *y0*, *@F* – дескриптор ODE-функції. Кожен рядок у масиві розв'язків *Y* відповідає значенню часу, що повертається у векторі-стовпці *T*;

- $[T, Y] = \text{solver} (@F, \text{tspan}, y0, \text{options})$ – дає розв'язок, подібний описаному вище, але з параметрами, визначеними значеннями аргументу *options*, створеного функцією *odeset* [4-7].

2.7.3.2. Технологія розв'язання диференціальних рівнянь у системі MATLAB така:

1. Створення m-файлу. Незалежно від виду системи він має вигляд

```
function dy = solverDE(t, y)
dy = zeros(n, 1);
dy(1) = f1 (t, y(1), y(2), ..., y(n));
dy(2) = f2 (t, y(1), y(2), ..., y(n));
.....
dy(n) = fn (t, y(1), y(2), ..., y(n)).
```

2. Отримання розв'язку і супроводжуючий його графік:

```
>> [T, Y] = solver('solverDE', [t0 tfinal], [y10 y20 ... yn0]);
>> plot(T, Y).
```

Приклад 2. Розв'язати диференціальне рівняння

$$y''' - 2y'' - y' + 2y = 0 \quad (2.4)$$

з одиничними початковими умовами.

Це диференціальне рівняння другого порядку приведемо до системи диференціальних рівнянь першого порядку:

$$\begin{cases} \frac{dy_1}{dt} = y_2; \\ \frac{dy_2}{dt} = y_3; \\ \frac{dy_3}{dt} = 2y_3 + y_2 - 2y_1, \end{cases} \quad (2.5)$$

із початковими умовами $y_1(0)=1, y_2(0)=1, y_3(0)=1$.

Вектор dy/dt правих частин системи рівнянь обчислюємо за допомогою власної функції **ex21** (рис. 2.71, а).

Тепер можна викликати функцію **ode45**, яка знаходить розв'язок нашої системи диференціальних рівнянь із початковими умовами $[1,1,1]$ на відрізку $[0,20]$ (рис. 2.71, б, в).

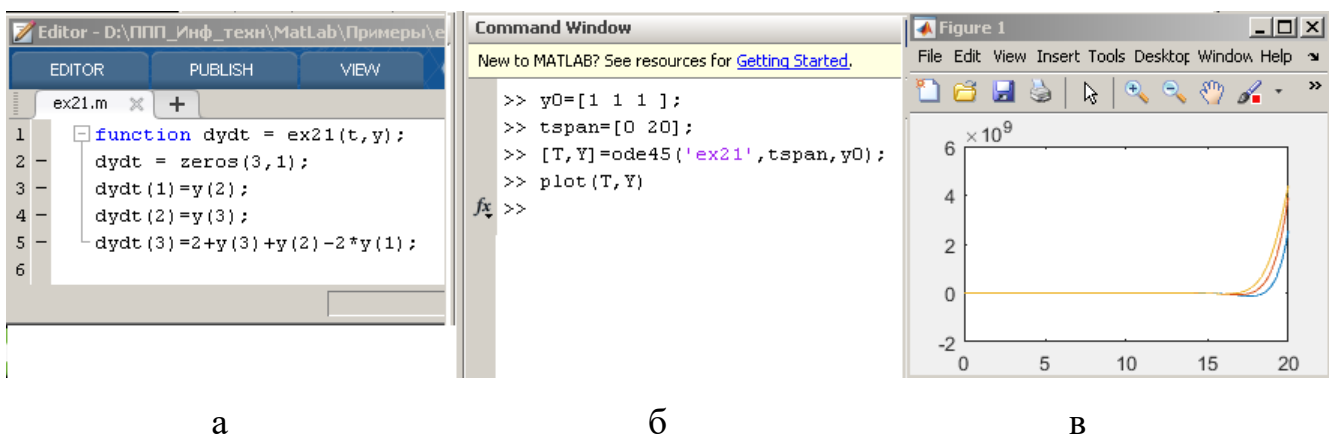


Рис. 2.71. Розв'язання прикладу 1:

а – створений m-файл; б – програма розв'язку; в – результат

2.7.3.3. Алгоритм розв'язання системи диференціальних рівнянь (СДР) розглянемо на прикладі розв'язання рівняння Ван-дер-Поля, записаного у вигляді системи з двох диференціальних рівнянь:

$$\begin{cases} \frac{dy_1}{dt} = y_2; \\ \frac{dy_2}{dt} = 100 \cdot (1 - y_2)^2 \cdot y_2 - y_1, \end{cases}.$$

Цю СДР потрібно розв'язати на часовому проміжку від 0 до 2 часових одиниць і за початкових умов $y_1(0) = 0$, $y_2(0) = 1$.

Порядок розв'язання такий:

➡ 1) записуємо систему диференціальних рівнянь у вигляді файл-функції. Для цього створюємо m-File з назвою `vdp100.m` (рис. 2.72, а).

Необхідно звернути увагу на те, що перед записом у тексті файл-функції рядків, які описують самі диференціальні рівняння, необхідно задати вектор-стовпець із нульових елементів. Кількість цих елементів відповідає кількості рівнянь. У наведеному прикладі цей вектор називають **dydt**. Також зазначимо, що назва кожного з диференціальних рівнянь обов'язково має починатися з назви вектора-стовпця, і далі в дужках вказують номер рівняння [4-7];

➡ 2) тепер чисельно розв'язуємо вказану СДР вводячи в командному вікні MATLAB відповідний запис. Методом чисельного розрахунку прийнятий **ode15s** (рис. 2.72, б).

```
>>[T,Y]=ode15s(@vdp100,[0 2],[0 1]);
```

➡ 3) побудуємо графік отриманої залежності (рис. 2.72 - в):

```
>> plot(T,Y), grid.
```

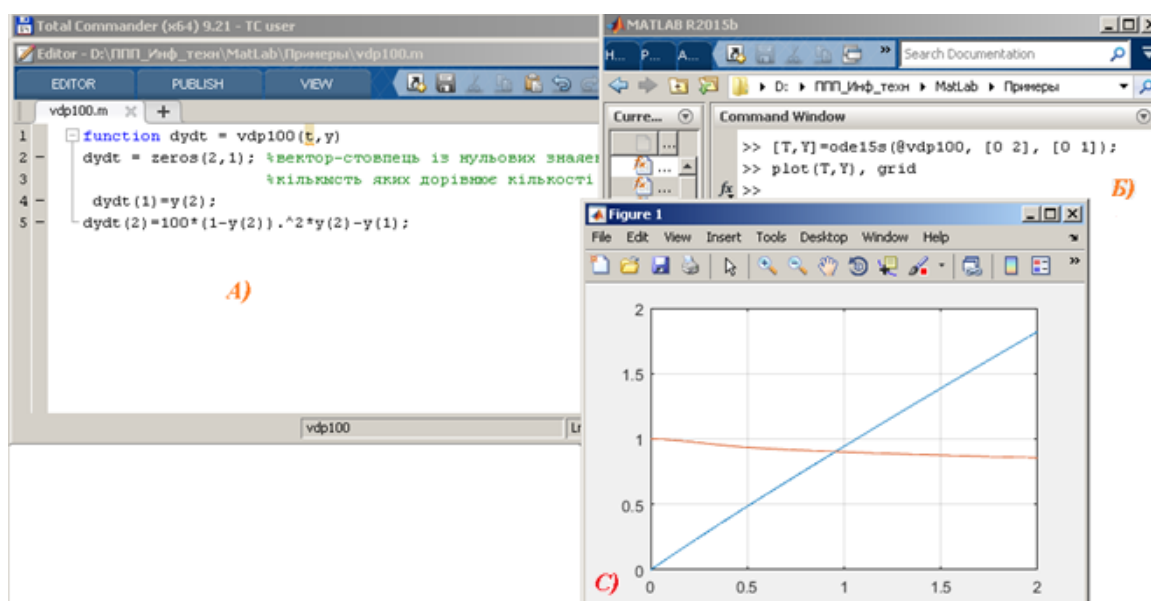


Рис. 2.72. Приклад розв'язання СДР

ЗАВДАННЯ ДЛЯ ВИКОНАННЯ

Завдання 1. За допомогою інтерполяції знайти значення таблично заданої функції в зазначених точках. Варіанти подано в табл.2.37.

Таблиця 2.37

Варіанти завдання 1

Варіант	Таблична функція для інтерполяції і апроксимації								
1	X	0	1	2	3	4	5	6	7
	Y	1	2.5	3	5	5.5	7	8	9
2	X	0	1	2	3	4	5	6	8
	Y	-1	-0.5	1	5	6.5	9	11.5	15
3	X	0	1	2	3	4	5	7	9
	Y	0.5	1	2.5	5	8.5	9.5	15	17.5
4	X	0	2	4	6	8	10	12	14
	Y	1	4	9	13	15	21	23	29
5	X	0	3	6	9	12	15	18	21
	Y	-2	8	16	25	35	45	52	62
6	X	0	2	4	6	10	16	20	25
	Y	-22	-14	-2	6	15	25	35	50
7	X	0	1	2	3	4	5	6	7
	Y	2	4	5	6	6.5	7	7.5	8
8	X	0	1	2	3	4	5	6	7
	Y	1	2.5	7	14.5	25	38.5	55	74.5
9	X	0	1	2	3	4	5	6	7
	Y	-1	0	3	8	15	24	35	48
10	X	0	1	2	3	4	5	6	7
	Y	-2	0	0.82	1.46	2	2.47	2.9	3.29

Завдання 2. Виконати апроксимацію таблично заданої функції. Варіант наступний за вашим у табл. 2.37.

Завдання 3. Знайти розв'язок звичайного диференціального рівняння першого порядку, зобразити графік результатів (задача Коші). Варіанти в табл. 2.38.

Варіанти завдання 3

Варіант	Диференціальне рівняння	Варіант	Диференціальне рівняння
1	$2xy'(x+1)-1=0, \quad y(1)=-1$	2	$y' + 2xy = e^{-x^2}, \quad y(0)=-1$
3	$xy' = y + \sqrt{xy}, \quad y(0)=1$	4	$y' + x^{-2} = e^{1/x}, \quad y(-1)=0$
5	$xy' = y - xe^{1/x}, \quad y(1)=0$	6	$xy' - 2y = x^3 \cos x, \quad y(\pi/6)=0$
7	$2yy'(1+e^x) = e^{2x}, \quad y(0)=1$	8	$x(y' - y) = (x+1)e^x, \quad y(1)=e$
9	$2y'x^2 = (x+y)^2, \quad y(1)=1$	10	$(y'x - y)\cos^2 x = x^2, \quad y(\pi/4)=0$

Завдання 4. Скласти програму чисельної інтеграції диференціальних рівнянь методом `ode15s`, яка забезпечувала б чисельне моделювання поведінки заданої системи за заданих умов.

При цьому забезпечити:

а) діалогове введення на екран значень усіх початкових параметрів, а саме часового діапазону та початкових значень $x_1(0)$, $x_2(0)$;

б) виведення графіків.

Задана система описує динаміку об'ємів популяцій хижаків $x_1(t)$ і жертв $x_2(t)$ і відомі як модель Вольтера.

$$\begin{cases} \frac{dx_1}{dt} = -a_{11}x_{11} + a_{12}x_1x_2 \\ \frac{dx_2}{dt} = a_{22}x_1 + a_{21}x_1x_2 \end{cases}.$$

Для всіх варіантів прийняти часовий діапазон – $[2, 10]$ часових одиниць; початкові значення $x_1(0)=2$, $x_2(0)=3$.

Коефіцієнти в системі рівнянь для кожного варіанта наведені в табл. 2.39.

Таблиця 2.39

Варіанти завдання 4

Варіант	Коефіцієнт рівняння			
	a_{11}	a_{12}	a_{22}	a_{12}
1	0,4	0,3	-0,6	0,7
3	0,3	0,35	-0,5	0,17
5	0,14	-0,3	0,65	-0,7
7	0,43	0,31	-0,26	0,27
9	0,34	-0,37	0,16	-0,4

Варіант	Коефіцієнт рівняння			
	a_{11}	a_{12}	a_{22}	a_{12}
2	0,3	0,35	-0,1	0,8
4	-0,4	0,2	0,6	0,3
6	-0,2	0,4	0,6	-0,1
8	-0,3	-0,33	0,1	0,7
10	0,35	0,33	-0,5	0,17

Завдання 5. Знайти розв'язок диференціального рівняння (задачі Коші) $\frac{dy}{dx} = f(x, y)$ на відрізку $[0, 2\pi]$ із початковою умовою $y(0) = y_0$.

Варіанти подані в табл. 2.40.

Таблиця 2.40

Варіанти завдання 5

Варіант	$f(x, y)$	y_0
1	$x^3 \sin y + 1$	0,0
3	$e^{y+x} + 3y$	2,0
5	$\sqrt{y^3 + x^2}$	0,4
7	$1/(1 * y^2) + xy$	0,1
9	$x^2 \cos y + 0,1$	0,3

Варіант	$f(x, y)$	y_0
2	$x^2 \sin y - 1$	0,1
4	$\sqrt{y^2 + x^3}$	0,3
6	$1/(1 * y^2) + x^2$	0,0
8	$\cos y + xy$	0,2
10	$x^3 \cos y + 0,1$	0,4

2.8. Основні можливості пакета математичного моделювання SIMULINK

Ознайомлення з методикою розв'язання задач моделювання та основними можливостями пакета математичного моделювання SIMULINK 4.0.

2.8.1. Початкові відомості про систему Simulink

Програма **Simulink** є додатком до пакета **MATLAB**. Із моделюванням із використанням **Simulink** реалізовано принцип візуального програмування, відповідно до якого користувач на екрані з бібліотеки стандартних блоків створює модель пристрою і здійснює розрахунки [4-7].

Іншими словами, **Simulink** дає змогу створювати і редагувати комп'ютерні моделі динамічної системи і виконувати над ними комп'ютерні експерименти.

Динамічна модельована система подана у вигляді функціональної блок-схеми, яку називають **Simulink-моделлю**, **S-моделлю** або просто **моделлю**. Створений файл має розширення **.mdl**.

Simulink є досить самостійним інструментом **MATLAB**, і працюючи з ним зовсім не потрібно знати сам **MATLAB** та інші його додатки. З іншого боку, доступ до функцій **MATLAB** та інших його інструментів залишається відкритим, і їх можна використовувати в **Simulink** [4-7].

Працюючи із **Simulink**, є можливість модернізувати бібліотечні блоки, створювати свої власні, а також становити нові бібліотеки блоків.

Моделюючи, можна вибирати метод розв'язання диференціальних рівнянь, а також спосіб зміни модельного часу (із фіксованим або змінним кроком). Під час моделювання є можливість стежити за процесами, що відбуваються в системі. Для цього використовують спеціальні пристрої спостереження, що входять до складу бібліотеки **Simulink**. Результати моделювання можуть бути подані у вигляді графіків або таблиць.

Середовище розроблення – це набір інструментів і засобів, які допомагають використовувати компоненти **Simulink**, а також підтримують зв'язок **Simulink** із користувачем і системою MATLAB [4-7].

Середовище розроблення **Simulink** включає:

- ➔ браузер універсальної та спеціалізованої бібліотек блокових компонентів **Simulink Library Browser**. Блоки організовані в палітри компонентів (набори) за призначенням;
- ➔ вбудований редактор блок-схем, заснований на графічному інтерфейсі користувача і є типовим засобом візуально-орієнтованого програмування;
- ➔ вікно створюваної **S**-моделі;
- ➔ засоби проектування на основі модельних уявлень;
- ➔ інструментарій для виконання **S**-моделі;
- ➔ інструментарій для аналізу;
- ➔ графічні вікна для виведення результатів і анімації **S**-моделі;
- ➔ браузер довідкової системи **Simulink Help**.

2.8.2. Запуск Simulink

Для запуску програми необхідно попередньо запустити пакет **MATLAB**.

Після відкриття основного вікна програми **MATLAB** потрібно запустити програму **Simulink**. Це можна зробити одним із трьох способів:

- натиснути кнопку  (**Simulink**) на панелі інструментів командного вікна **MATLAB**;
- у командному рядку головного вікна **MATLAB** набрати команду **simulink (>>simulink)** і натиснути клавішу **Enter** на клавіатурі;
- виконати команду **Open...** у меню **File** і відкрити файл моделі (**mdl** - файл).

Останній варіант зручно використовувати для запуску вже готової і налагодженої моделі, коли потрібно лише провести розрахунки і не потрібно додавати нові блоки в модель. Використання першого і другого способів призводить до відкриття вікна оглядача розділів бібліотеки **Simulink** (рис. 2.73).

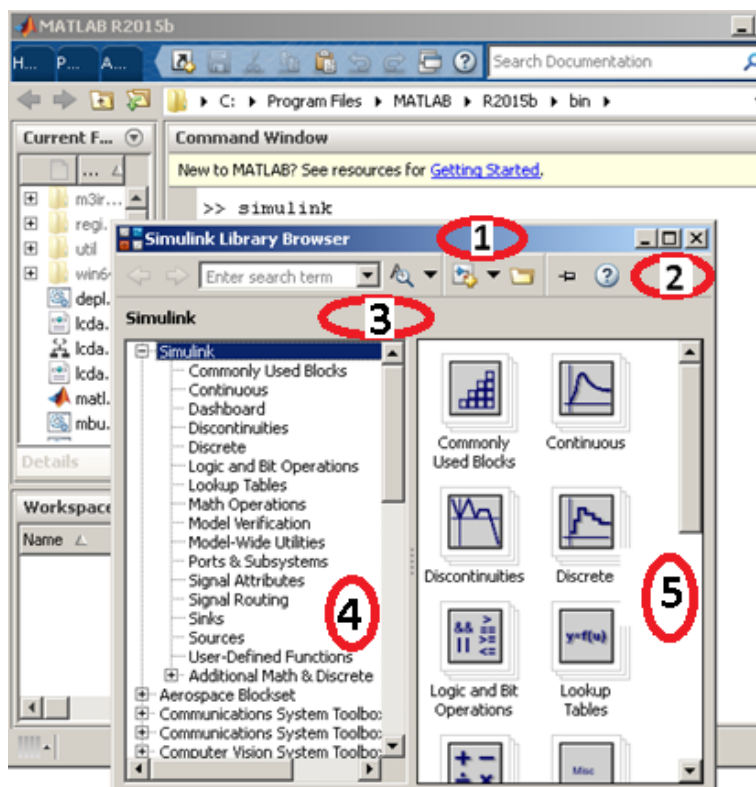


Рис. 2.73. Основне вікно **MATLAB** з оглядачем бібліотеки розділів **Simulink**

При цьому не створюють і не завантажують S-модель.

Вікно браузера (оглядача) бібліотек блокових компонентів містить такі елементи:

1. Заголовок вікна.
2. Панель інструментів.
3. Коментар повідомлень про вибраний блок.
4. Список розділів бібліотеки, який реалізований у вигляді дерева [4-7].

Список розділів бібліотеки **Simulink** поданий у вигляді дерева, і правила роботи з ним є загальними для списків такого виду:

- піктограма згорнутого вузла дерева містить символ «+», а піктограма розгорнутого містить символ «-»;

- для того щоб розгорнути або згорнути вузол дерева, достатньо натиснути на його піктограмі лівою клавішею миші (ЛКМ).

5. Вікно вмісту розділу поточної бібліотеки (список блоків).

На рис. 2.73 виділена основна бібліотека **Simulink** (у лівій частині вікна) і показані її розділи (у правій частині вікна). Бібліотека **Simulink** містить такі основні розділи:

1. **Continuous** – лінійні блоки.
2. **Discrete** – дискретні блоки.
3. **Functions & Tables** – функції та таблиці.
4. **Math** – блоки математичних операцій.
5. **Nonlinear** – нелінійні блоки.
6. **Signals & Systems** – сигнали і системи.
7. **Sinks** – реєструючі пристрої.
8. **Sources** – джерела сигналів і впливів.
9. **Subsystems** – блоки підсистем тощо.

2.8.3. Створення (побудова) S-моделі

Щоб створити свою програму, необхідно створити новий файл моделі (із розширенням **.slx**), розташувати блоки у вікні моделі. Для цього відкрити відповідний розділ бібліотеки. Далі, указавши курсором на необхідний блок і натиснувши на ліву клавішу миші, «перетягнути» блок у створене вікно [4-7].

Створюють нову S-модель одним із способів:

- ➡ комбінацією клавіш **Ctrl + N**;
- ➡ вибором кнопки  на панелі інструментів;

➡ вибором команди **File → New → Model** у меню командного вікна системи МАТЛАБ або меню браузера бібліотеки компонентів Simulink Library Browser.

Для створення моделі в середовищі **Simulink** необхідно послідовно виконати ряд дій.

2.8.3.1. Створити новий файл моделі за допомогою одним із вищезгаданих способів (наприклад команди **New Model** або **File / New / Model** (рис. 2.74)). Новостворене вікно моделі показано на рис. 2.75.

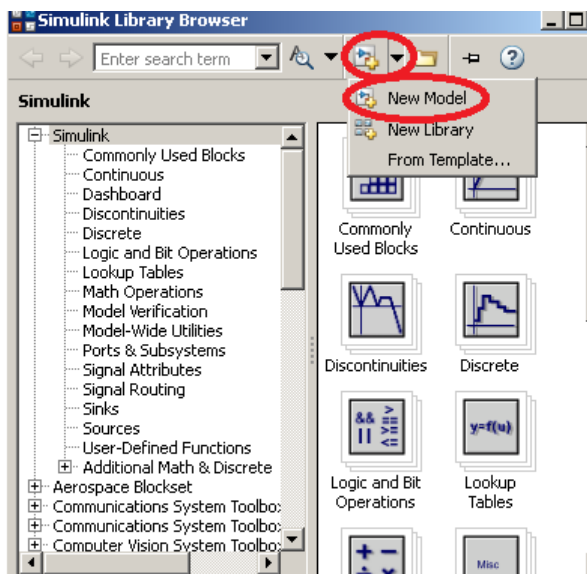


Рис. 2.74. Створення нової моделі

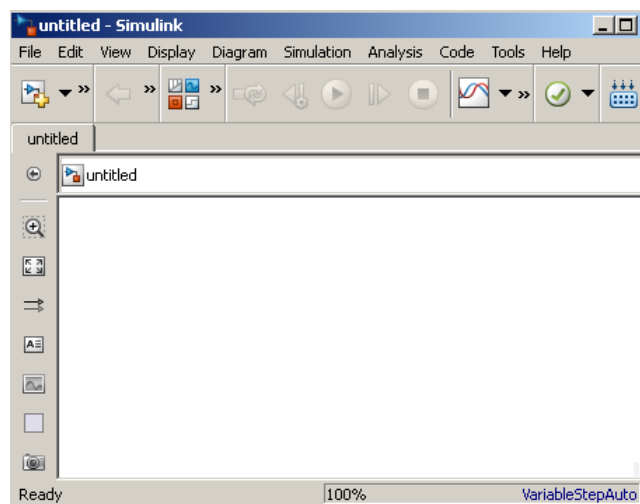


Рис. 2.75. Порожнє вікно моделі

2.8.3.2. Розташувати блоки у вікні моделі. Для цього необхідно відкрити відповідний розділ бібліотеки (наприклад **Commonly Used Blocks** – **Загальновживані блоки** (або **Sources** – **Джерела**)). Далі, вказавши курсором на необхідний блок і натиснувши на ліву клавішу миші, «перетягнути» блок у створене вікно. **Клавішу миші потрібно тримати натиснутою.** На рис. 2.76 показано вікно моделі, що містить три блоки. Стрілками показано, у якій бібліотеці знаходяться ці блоки.

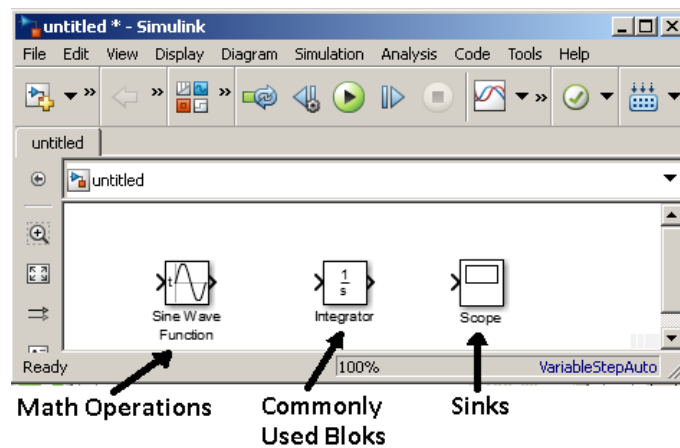


Рис. 2.76. Вікно моделі, що містить блоки

Для видалення блока необхідно вибрати блок (вказати курсором на його зображення і натиснути ліву клавішу миші), а потім натиснути клавішу **Delete** на клавіатурі [4-7].

Для зміни розмірів блока потрібно вибрати блок, встановити курсор в один із кутів блока і, натиснувши ліву клавішу миші, змінити розмір блока (курсор при цьому перетвориться на двосторонню стрілку).

2.8.3.3. Далі, якщо це потрібно, можна змінити параметри блока, встановлені програмою «за замовчуванням». Для цього необхідно двічі натиснути ліву клавішу миші, вказавши курсором на зображення блока. Відкриється вікно редагування параметрів цього блока. Задаючи чисельні параметри, потрібно розуміти, що як десятковий роздільник використано крапку, а не кому. Після внесення змін потрібно закрити вікно кнопкою ОК. На рис. 2.77 для прикладу показаний блок, що моделює передавальну функцію і вікно редагування параметрів цього блока.

2.8.3.4. Після установалення на схемі всіх блоків із необхідних бібліотек потрібно з'єднати елементи схеми.

Кожен блок має кутові дужки (>) з одного або декількох боків (рис. 2.78).

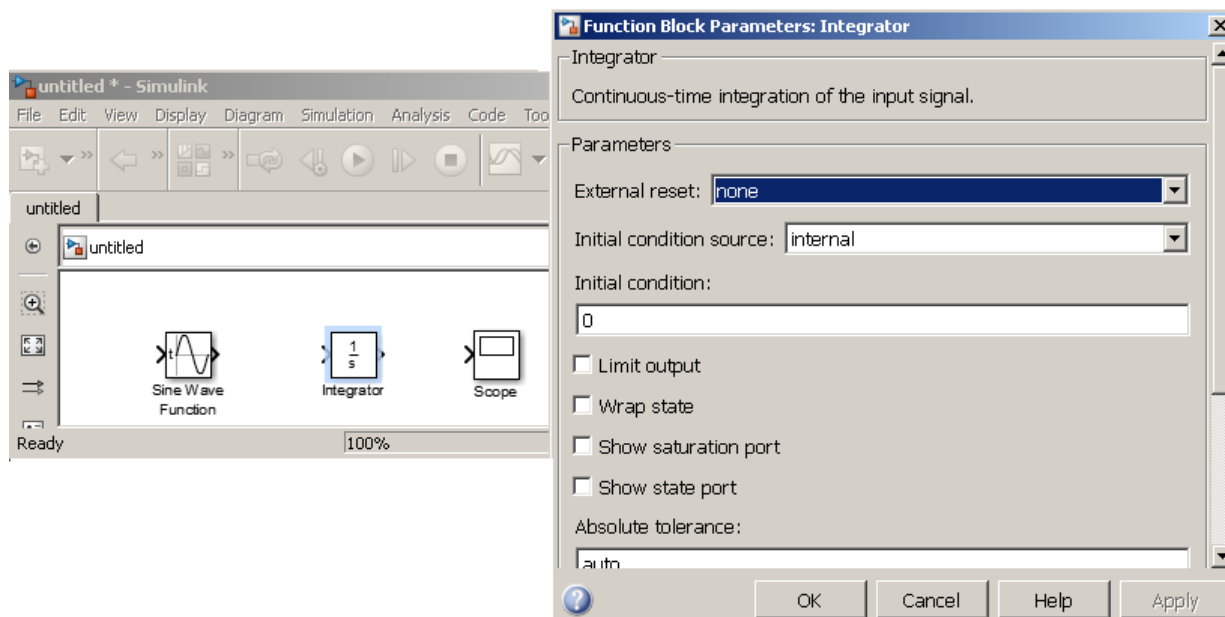


Рис. 2.77. Блок, що моделює передавальну функцію, і вікно редагування параметрів блока

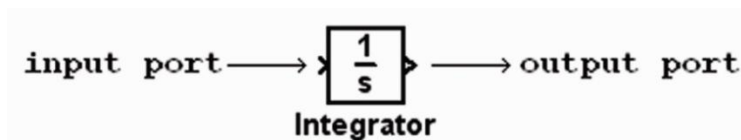


Рис. 2.78. Параметри інформаційного блока

Символ >, що входить у блок, **називають** вхідним портом, або **входом**.
Символ >, що виходить із блока, **називають** вихідним портом, або **виходом**.

Для з'єднання блоків (*інформаційних зв'язків*) необхідно вказати курсором на «вихід» блока, а потім натиснути і, не відпускаючи ліву клавішу миші, провести лінію до входу іншого блока. Після чого відпустити клавішу. У разі правильного з'єднання зображення стрілки на вході блока змінює колір [4-7].

Для створення точки розгалуження в сполучній лінії потрібно підвести курсор до передбачуваного вузла і, натиснувши праву клавішу миші, протягнути лінію. Для видалення лінії потрібно вибрати лінію (так само, як це виконано для блока), а потім натиснути клавішу **Delete** на клавіатурі. Схема моделі, у якій виконані з'єднання між блоками, показана на рис. 2.79.

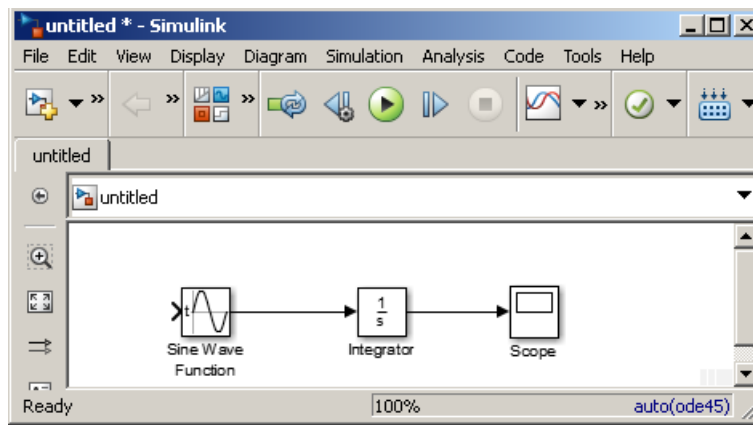


Рис. 2.79. Схема моделі

2.8.3.5. Після складання розрахункової схеми необхідно зберегти її у вигляді файлу на диску, вибравши пункт меню **File / Save As...** у вікні схеми і вказавши папку та ім'я файлу (із розширенням **.slx**). Потрібно розуміти, що ім'я файлу не має перевищувати 32 символів, має починатися з літери і не може містити символи кирилиці і спецсимволи. Ця вимога стосується і шляху файлу (до тих папок, у яких зберігають файл). Для подальшого редагування схеми можна користуватися пунктом меню **File / Save**. Із повторними запусками програми SIMULINK завантаження схеми здійснюється за допомогою меню **File / Open...** у вікні оглядача бібліотеки або з основного вікна MATLAB [4-7].

2.8.4. Управління роботою S-моделі

Запускають S-модель на виконання (запуск розрахунку) через вибір пункту меню **Simulation → Start**, комбінацію клавіш **Ctrl + T** або піктограми  на панелі інструментів.

Робота S-моделі (виконання розрахунку) триватиме до закінчення інтервалу часу моделювання.

Процес можна завершити достроково, вибравши пункт меню **Simulation → Stop** або натиснувши на піктограму .

Виконання S-моделі також можна призупинити, вибравши пункт меню **Simulation→Pause** або піктограму , і потім продовжити за допомогою пункту меню **Simulation→Continue** або піктограми .

2.8.5. Завершення роботи S-моделі

Для завершення роботи необхідно:

- ➡ дочекатися закінчення або зупинити виконання S-моделі;
- ➡ зберегти S-модель у файлі;
- ➡ закрити вікно S-моделі, вікно браузера бібліотеки блокових компонентів Simulink, а також головне вікно системи MATLAB [4-7].

2.8.6. Налашдування і виконання створеної моделі

1. Налаштувати блок генератора синусоїдальної напруги **Sine Wave** (за основу взята модель рис. 2.79). На піктограмі блока зробимо двічі натиснемо ліву кнопку миші і в меню змінимо параметри (рис. 2.80).

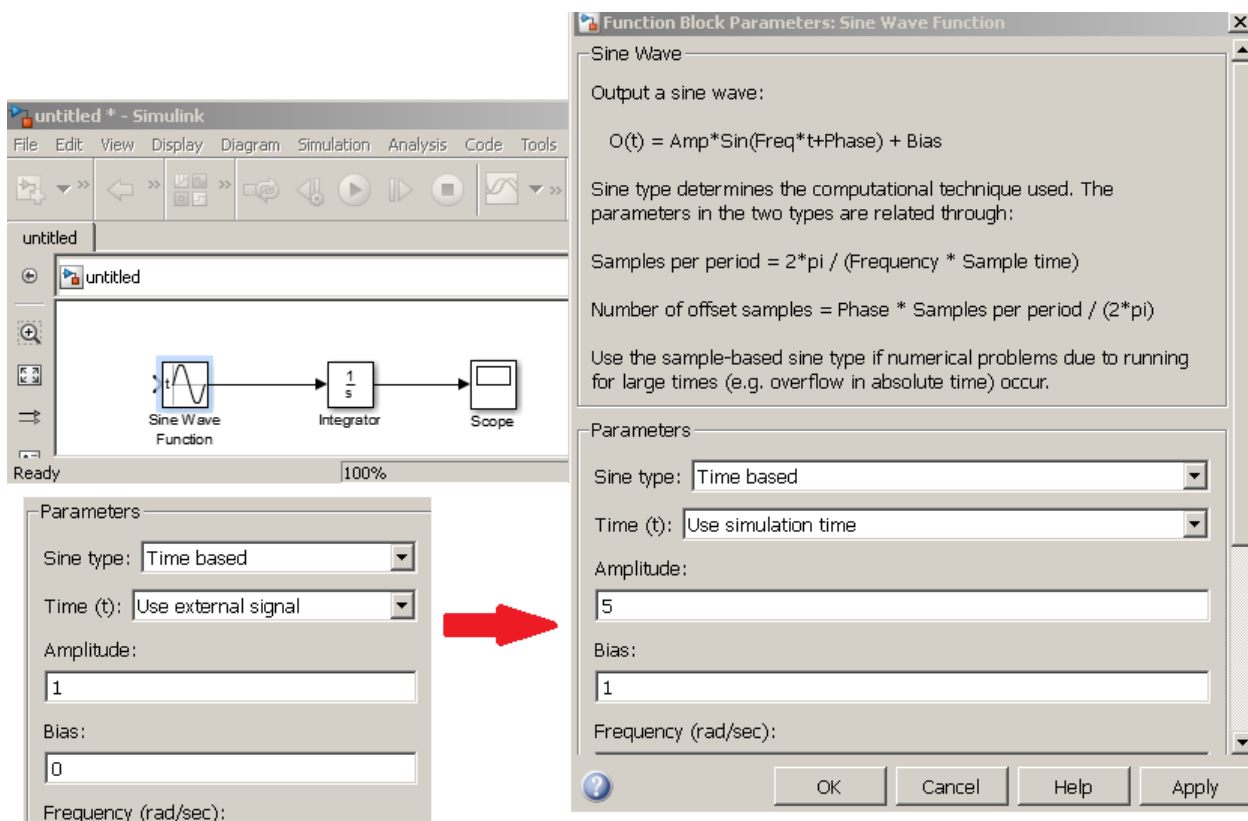


Рис. 2.80. Задавання параметрів блока

2. Зберегти S-модель файл на магнітному диску (**File / Save As ...** , задавши ім'я файлу, наприклад **Sprim2**) і запустити модель на виконання (рис. 2.81).

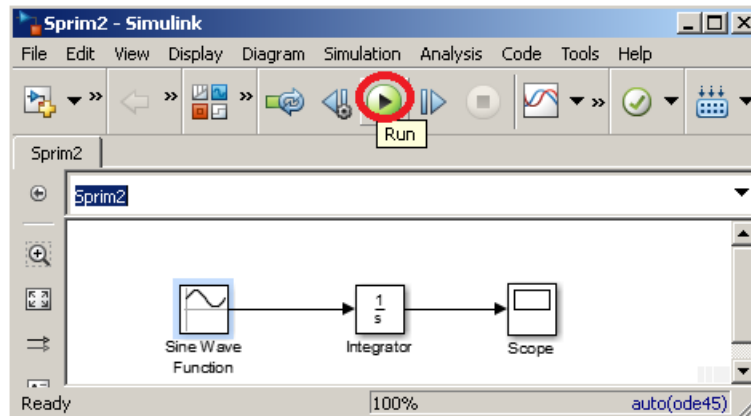


Рис. 2.81. Запуск моделі на виконання

3. Після виконання моделі подивитися створений графік функції. Для цього двічі натиснути на блоці **Scope**. Результат поданий на рис. 2.82.

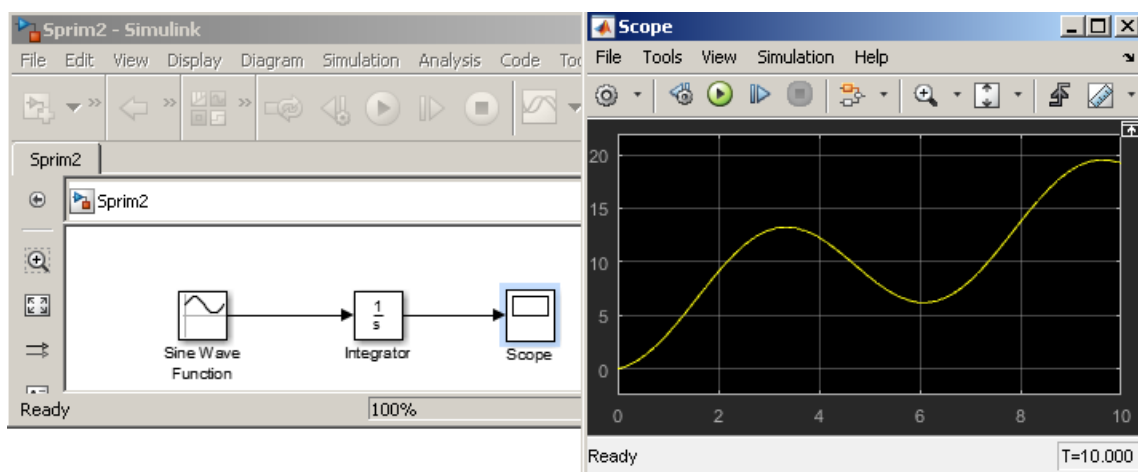


Рис. 2.82. Результат виконання створеної моделі

2.8.7. Приклади розв'язання найпростіших задач моделювання

Приклад 1. Мальтусівська жорстка модель динаміки приросту популяції особин деякого виду описана звичайним диференціальним рівнянням

$$\frac{dN}{dt} = kN,$$

де $N(t)$ – чисельність особин популяції;

k – коефіцієнт приросту популяції.

Побудувати криву приросту популяції за умови, що $k = 2$, $N_0 = 0.1$, $t_{sim} = 2$ с, де N_0 – початкова чисельність особин популяції, t_{sim} – час симуляції.

Розв'язання:

1. Створимо схему розв'язання рівняння у вигляді нової S-моделі (рис. 2.83, а).
2. Розкриємо блок Gain, задамо коефіцієнт $k = 2$ і блок інтегратора, початкове значення $N_0 = 0.1$.
3. Виконаємо модель і після закінчення моделювання, розкриваючи блок Scope, можна побачити графік, що демонструє динаміку приросту популяції (рис. 2.83, б).
4. Установити час симуляції (по осі X на графіку) можна через **Simulation/ Model Configuration Parameters** або натисканням **Ctrl+E**.

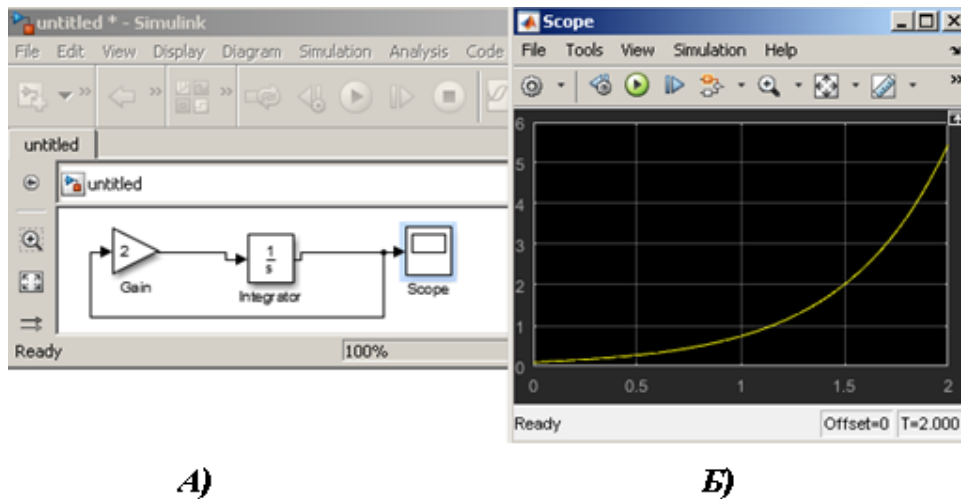


Рис. 2.83. Модель розв'язання прикладу 1 і растровий графік динаміки приросту популяції $N(t)$

Приклад 2. Логістична модель динаміки приросту популяції описана звичайним диференціальним рівнянням

$$\frac{dN}{dt} = aN - bN^2,$$

де $N(t)$ – чисельність особин популяції;

a – коефіцієнт народжування;

b – коефіцієнт смертності.

Побудувати логістичну криву приросту популяції, якщо початкове значення щільності особин $N_0 = 0.3$, коефіцієнт народжуваності $a = 10$, коефіцієнт смертності $b = 20$, час симуляції $t_{sim} = 1$ с.

Розв’язання:

1. Створимо схему розв’язання рівняння (рис. 2.84, а) і, розкривши блок інтегратора, задамо початкове значення $N_0 = 0.3$.

2. Після закінчення моделювання, розкриваючи блок **Scope**, можна побачити графік, що демонструє динаміку приросту популяції $N(t)$ (рис. 2.84, б).

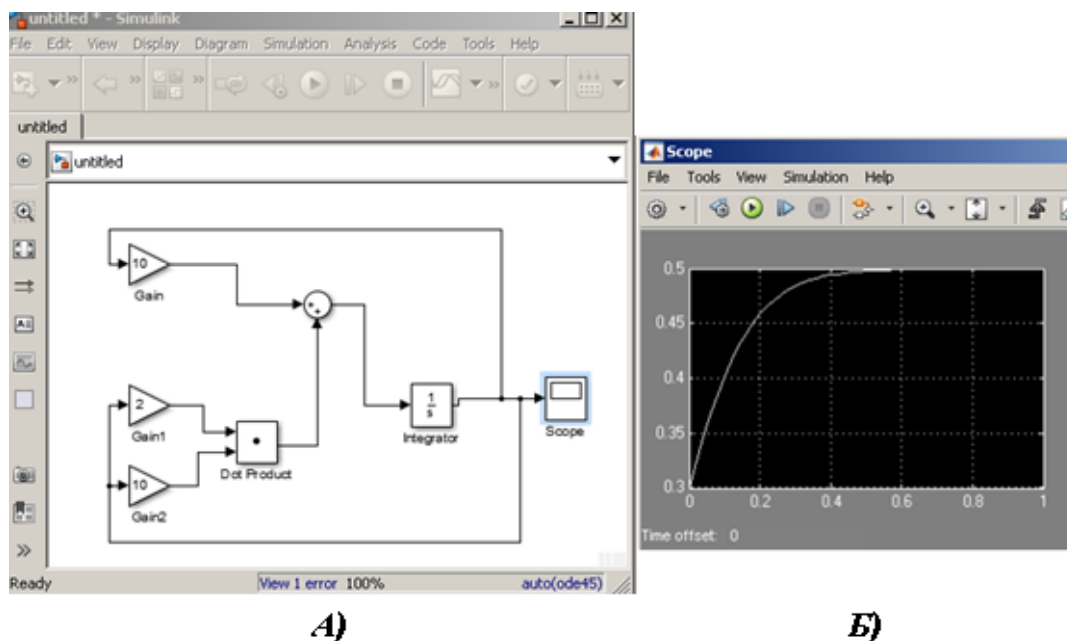


Рис. 2.84. Блок-схема розв’язання прикладу 2 і логістична крива приросту популяції $N(t)$

Приклад 3. У місті з 20000 жителів з'являються 50 інфекційних хворих, що викликає епідемію. Припустимо, що приріст хворих людей за день пропорційний кількості контактів хворих і здорових людей, тобто добутку кількості здорових (тих, хто ще не хворів і не здобув імунітет) і кількості хворих.

Коефіцієнт пропорційності (він характеризує швидкість поширення епідемії) приймемо рівним 10-4. Як розвивається епідемія, як день у день змінюється кількість хворих людей? На який день буде максимальна кількість хворих?

Розв'язання

Позначимо через x кількість хворих, через y – кількість здорових, через k – коефіцієнт пропорційності.

Математична модель розвитку епідемії описана системою звичайних диференціальних рівнянь другого порядку

$$\begin{cases} \dot{x} = -kxy \\ \dot{y} = kxy - y' \end{cases}$$

де $x(0) = 50$, $y(0) = 20000$, $k = 0,0001$.

Будуємо блок-схему розв'язання задачі в системі Simulink (рис. 2.85, а) і, розкривши блок інтегратора, задамо початкові значення: $x(0) = 2$ (параметр $y1$), $y(0) = 0.01$ (параметр $y2$). Після закінчення моделювання, розкриваючи блоки Scope і XY-Graph, можна побачити графіки зміни чисельності хворих і здорових (відповідно) людей у місті (рис. 2.85, б).

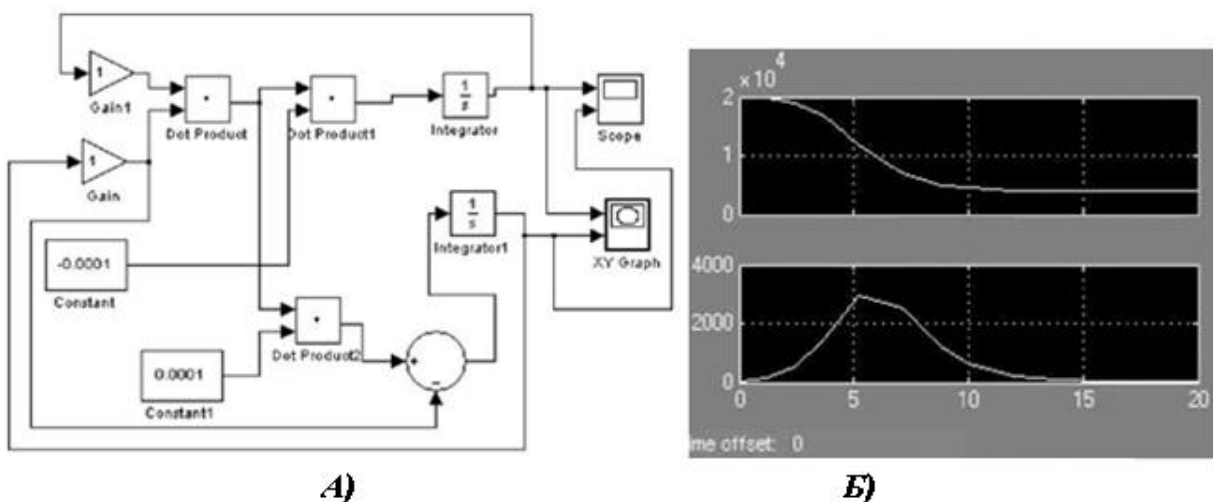


Рис. 2.85. Схема розв'язання задачі прикладу 3 і растрові графіки $x(t)$, $y(t)$

ЗАВДАННЯ ДЛЯ ВИКОНАННЯ

Завдання 1. Виконати приклади 1, 2, 3 із теоретичних відомостей.

Завдання 2. Використовуючи модель прикладу 2 (рис. 2.84) побудувати логістичну криву приросту популяції, якщо початкове значення щільності особин N_0 , коефіцієнт народжуваності a , коефіцієнт смертності b , час симуляції t_{sim} , с, наведені в табл. 2.41.

Таблиця 2.41

Варіанти завдань

Варіант	N_0	a	b	t_{sim}
1	5	20	25	3
3	7	22	20	4
5	9	25	21	5
7	11	21	18	3
9	13	18	28	4

Варіант	N_0	a	b	t_{sim}
2	7	12	12	2
4	4	13	13	3
6	8	15	12	4
8	9	17	13	5
10	3	11	14	2

Зверніть увагу, як істотно змінюється характер кривої, що описує динаміку зміни популяції. Така відмінність обумовлена відношенням параметрів a і b до початкового значення N_0 . Якщо $a/b > N_0$, то логістична

крива, що описує динаміку приросту популяції, наближається до значення a/b знизу, якщо відношення $a/b < N_0$, то щільність популяції зменшується, і логістична крива прагне до a/b зверху.

Завдання 3. Використовуючи модель прикладу 3 (рис. 2.85) побудувати графіки зміни чисельності хворих $x(t)$ і здорових $y(t)$ людей у місті, використовуючи дані табл. 2.42, де x – кількість хворих, y – кількість здорових, k – коефіцієнт пропорційності.

Таблиця 2.42

Варіанти завдань

Варіант	$x(0)$	$y(0)$	k
1	45	25000	10^{-4}
3	55	30000	10^{-3}
5	60	40000	10^{-4}
7	100	100000	10^{-3}
9	150	150000	10^{-4}

Варіант	$x(0)$	$y(0)$	k
2	300	800000	10^{-4}
4	350	1000000	10^{-3}
6	400	1500000	10^{-5}
8	370	1000000	10^{-4}
10	150	1200000	10^{-4}

Бібліографічний список

1. Паранчук Я. С., Мороз В. І. Обчислювання та програмування. MathCAD: навч. посіб. Вид. 2-ге. Львів: Видавництво Львівської політехніки, 2013. 364 с.
2. Дзісь В. Г., Левчук О. В., Новицька Л. І., Смілянець О. Г., Бубновська І. А. Довідник. Програмування в MathCAD. Вінниця. Видавничий центр ВНАУ, 2015. 187 с.
3. Ситніков О. В. Основи роботи з MathCAD: метод. вказівки до виконання комп'ютерного практикуму з дисципліни «Комп'ютерні технології та програмування – 2. Програмні засоби для числового аналізу», для студентів напряму підготовки «Автоматизація та комп'ютерно-інтегровані технології». Київ: НТУУ «КПІ», 2013. 103 с.
4. Забара С. С. Моделювання систем у середовищі MATLAB: навч. посіб. Київ: Університет «Україна», 2011. 176 с.
5. Коржик М. В. Моделювання об'єктів та систем керування засобами MATLAB: навч. посіб. для студ. вищ. навч. закл. Київ: НТУУ «КПІ», 2016. 174 с.
6. Лазарєв Ю. Ф. Початки програмування в середовищі MATLAB: навч. посіб. Київ: «Політехніка», 2000. 396 с.
7. Гоблик Н. М., Гоблик В. В. MATLAB в інженерних розрахунках. Комп'ютерний практикум: навч. посіб. Львів: Національний університет «Львівська політехніка», 2020. 190 с. URL: https://cions.kpi.ua/Arhiv/Lazarev/dovidnyk_Matlab.pdf.
8. Кузяєв І. М., Ситар В. І. Основи комп'ютерного моделювання технічних систем: навч. посіб. Дніпро: ДВНЗ УДХТУ, 2020. 392 с.
9. Вірченко Н. О., Ляшко І. І. Графіки елементарних та спеціальних функцій. Київ: Наук. думка, 1996. 582 с.

Навчальний посібник

Змій Сергій Олексійович

Лабенко Дмитро Петрович

Панченко Артем Сергійович

та ін.

СУЧАСНІ ПАКЕТИ MATHCAD I MATLAB
ДЛЯ ІМІТАЦІЙНОГО МОДЕЛЮВАННЯ
В ІНЖЕНЕРНІЙ ДІЯЛЬНОСТІ

Відповідальний за випуск Лабенко Д. П.

Редактор Ібрагімова Н. В.

Підписано до друку 15.06.2026 р.

Умовн.-друк.арк. 13,0. Тираж . Замовлення № .

Видавець і виготовлювач Український державний університет
залізничного транспорту,
61050, Харків-50, майдан Феєрбаха, 7.

Свідоцтво суб'єкта видавничої справи ДК № 6100 від 21.03.2018 р.