

ІНСТРУКТИВНО-МЕТОДИЧНІ МАТЕРІАЛИ

для виконання лабораторних робіт

з дисципліни

«Чисельні методи »

ЗІ СПЕЦІАЛЬНОСТІ 121

«ІНЖЕНЕРІЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ»

Розробив: Ланська С.С.

Розглянуто та схвалено

на засіданні циклової комісії

програмної інженерії

Протокол № 4 від 17.11.2021 р.

Голова комісії _____ Ланська С.С.

Дніпро,
2023

КРИТЕРІЇ ОЦІНЮВАННЯ ЗНАНЬ СТУДЕНТІВ

З ЛАБОРАТОРНИХ РОБІТ

З ДИСЦИПЛІНИ «ЧИСЕЛЬНІ МЕТОДИ»

зі спеціальності 121

«Інженерія програмного забезпечення»

Оцінка «5» ставиться, якщо:

- робота виконана повністю;
- у алгоритму рішення задачі та в теоретичних викладках рішення немає пропусків і помилок;
- у тексті звіту немає синтаксичних помилок (можливі одна-дві різні неточності, описки, які не є наслідком незнання чи нерозуміння навчального матеріалу).

Оцінка "4" ставиться, якщо:

- робота виконана повністю, але обґрунтування кроків вирішення недостатні (якщо вміння обґрунтовувати міркування не було спеціальним об'єктом перевірки);
- допущена одна помилка або два-три недоліки.

Оцінка «3» ставиться, якщо:

- допущені більше однієї помилки або двох-трьох недоліків в викладках, алгоритмі або тексті звіту, але учень володіє обов'язковими вміннями по перевірній темі.

Оцінка «2» ставиться, якщо:

- допущені суттєві помилки, які показали, що учень не володіє обов'язковими знаннями з даної теми в повній мірі.

ЛАБОРАТОРНА РОБОТА № 1

з навчальної дисципліни «Чисельні методи»

Тема Знайомство з системою комп'ютерної математики – математичною матричною лабораторією MATLAB

Мета Ознайомитися з основними елементами і складовими частинами системи комп'ютерної математики MATLAB і її робочим і програмним середовищем

Час – 4 години

Теоретичні відомості

1) Робоче середовище системи MATLAB.

При запуску програмного продукту MATLAB на екрані відкривається робоче вікно програми, зображене на рис. 1.1.

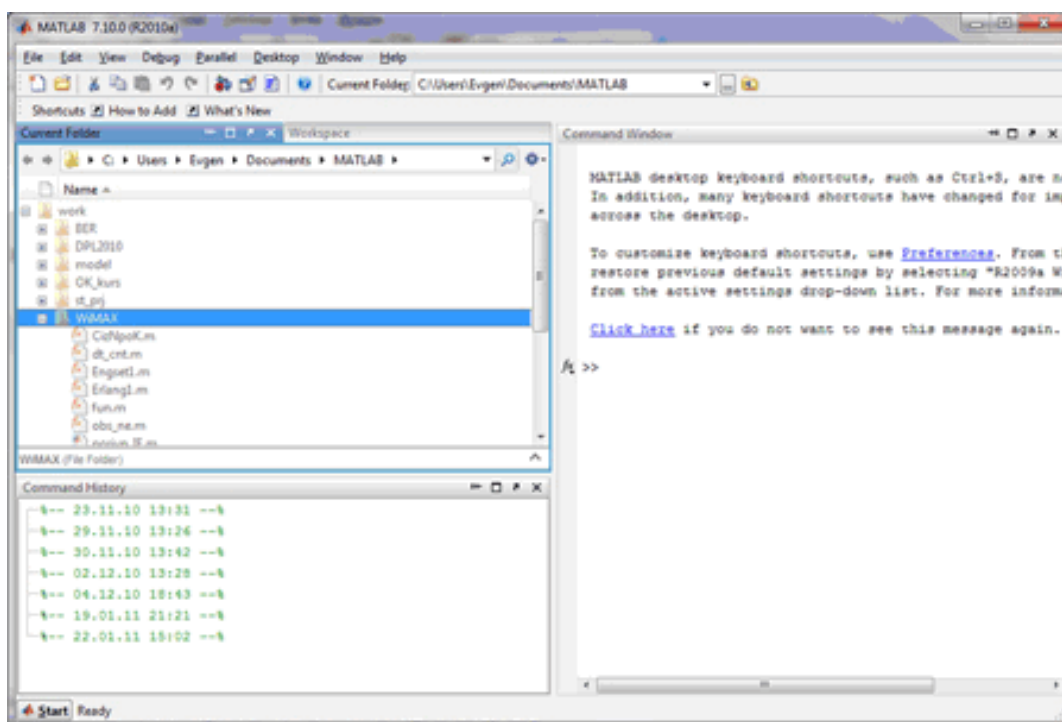


Рисунок 1.1 – Робоче середовище системи MATLAB

Основними елементами робочого середовища є:

- меню;
- панель інструментів з кнопками і випадаючим меню;

- вікна з вкладками *Workspace* і *Current Directory* для перегляду змінних та для встановлення робочого каталогу з використовуваними файлами-сценаріями і функціями;
- вікно *Command Window* (аналог командної стрічки), призначене для введення команд і виведення результатів;
- вікно *Command History* - для перегляду і повторного виконання раніше введених команд;
- рядок стану та кнопка *Start*.

При натисненні на кнопку **Start** відкривається меню, приклад якого зображено на рис. 1.2, з його допомогою забезпечується доступ до усіх основних засобів системи.

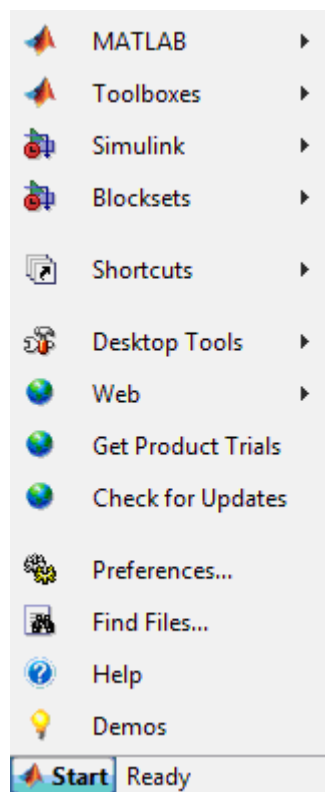


Рисунок 1.2 – Меню, яке відкривається при натисненні на кнопку Start

Вікно *Command Window* складається з наступних елементів:

- заголовка з назвою вікна і двома кнопками справа;
- робочої області з командною стрічкою, в якій знаходиться мигаючий вертикальний курсор;
- смуг прокрутки.

В полі назви кожного вікна, поряд з кнопкою закривання, знаходиться кнопка «**Undock...**» для витягування вікна з робочого середовища (якщо воно вбудовано), або кнопка «**Dock...**» для вбудовування окремого вікна в робоче середовище MATLAB.

При запуску пакету в робочій області вікна *Command Window* з'являються два посилання: *MATLAB Help* і *Demos* – для виклику довідкової системи або демонстраційних прикладів.

2) Арифметичні обчислення.

Вбудовані математичні функції дозволяють знаходити значення виразів будь-якої складності, причому для обчислень можна використовувати також мову сценаріїв системи, що є мовою програмування високого рівня.

Також система MATLAB надає можливість управління форматом виведення результатів.

Взагалі виведення проміжної інформації у командне вікно підпорядковується таким правилам:

- якщо запис оператора не закінчується символом ';', результат дії цього оператора одразу ж виводиться в командне вікно;

- якщо оператор закінчується символом ';', результат його дії не відображується в командному вікні;

- якщо оператор не містить знака присвоювання (=), тобто є просто записом деякої послідовності дій над числами і змінними, значення результату присвоюється спеціальній системній змінній за ім'ям ans;

- отримане значення змінної ans можна використовувати в наступних операторах обчислень, використовуючи це ім'я ans; при цьому варто пам'ятати, що значення системної змінної ans змінюється після дії чергового оператора без знака присвоювання;

- у загальному випадку форма подання результату в командне вікно має вид: <Ім'я змінної> = <результат>.

Система Matlab має кілька імен змінних, що використовуються самою системою і входять до складу зарезервованих:

- i, j – уявна одиниця (корінь квадратний з -1);

- π – число π (зберігається у виді 3.141592653589793);
- inf – позначення машинної нескінченності;
- NaN – позначення невизначеного результату (наприклад, типу 0/0 або inf/inf);

- eps – похибка операцій над числами із плаваючою комою;
- ans – результат останньої операції без знака присвоювання;
- realmax – максимальна величина числа, що може бути використана;
- realmin – мінімальна величина числа, що може бути використана.

Ці змінні можна використовувати в математичних виразах.

У мові Matlab передбачені наступні елементарні арифметичні функції.

Тригонометричні й гіперболічні функції:

- $\sin(Z)$ – синус числа Z ;
- $\sinh(Z)$ – гіперболічний синус;
- $\text{asin}(Z)$ – арксинус (у радіанах, у діапазоні від $-\pi/2$ до $+\pi/2$);
- $\text{asinh}(Z)$ – обернений гіперболічний синус;
- $\cos(Z)$ – косинус;
- $\cosh(Z)$ – гіперболічний косинус;
- $\text{acos}(Z)$ – арккосинус (у діапазоні від 0 до π);
- $\text{acosh}(Z)$ – обернений гіперболічний косинус;
- $\tan(Z)$ – тангенс;
- $\tanh(Z)$ – гіперболічний тангенс;
- $\text{atan}(Z)$ – арктангенс (у діапазоні від $-\pi/2$ до $+\pi/2$);
- $\text{atan2}(X,Y)$ – чотириквadrantний арктангенс (кут у діапазоні від $-\pi$ до $+\pi$ між горизонтальним правим променем і променем, що проходить через точку з координатами X і Y);
- $\text{atanh}(Z)$ – обернений гіперболічний тангенс;
- $\sec(Z)$ – секанс;
- $\text{sech}(Z)$ – гіперболічний секанс;
- $\text{asec}(Z)$ – арксеканс;
- $\text{asech}(Z)$ – обернений гіперболічний секанс;

- $\csc(Z)$ – косеканс;
- $\operatorname{csch}(Z)$ – гіперболічний косеканс;
- $\operatorname{acsc}(Z)$ – арккосеканс;
- $\operatorname{acsch}(Z)$ – обернений гіперболічний косеканс;
- $\cot(Z)$ – котангенс;
- $\operatorname{coth}(Z)$ – гіперболічний котангенс;
- $\operatorname{acot}(Z)$ – арккотангенс;
- $\operatorname{acoth}(Z)$ – обернений гіперболічний котангенс.

Експоненціальні функції:

- $\exp(Z)$ – експонента числа Z ;
- $\log(Z)$ – натуральний логарифм;
- $\log_{10}(Z)$ – десятковий логарифм;
- $\sqrt{Z}$ – квадратний корінь із числа Z ;
- $\operatorname{abs}(Z)$ – модуль числа Z .

Цілочислові функції:

- $\operatorname{fix}(Z)$ – округлення до найближчого цілого убік нуля;
- $\operatorname{floor}(Z)$ – округлення до найближчого цілого убік від'ємної нескінченності;
- $\operatorname{ceil}(Z)$ – округлення до найближчого цілого убік додатної нескінченності;
- $\operatorname{round}(Z)$ – звичайне округлення числа Z до найближчого цілого;
- $\operatorname{mod}(X, Y)$ – цілочислове ділення X на Y ;
- $\operatorname{rem}(X, Y)$ – обчислення остачі від ділення X на Y ;
- $\operatorname{sign}(Z)$ – обчислення сигнум-функції числа Z (0 при $Z=0$, -1 при $Z<0$, 1 при $Z>0$).

В арифметичних виразах мови MATLAB використовуються такі знаки арифметичних операцій:

- $+$ – додавання;
- $-$ – віднімання;
- $*$ – множення;

- / – ділення зліва праворуч;
- \ – ділення справа ліворуч;
- ^ – піднесення до степеня.

3) Найпростіші обчислення.

Виберіть вигляд робочого середовища „по замовчуванню”, наберіть в командному рядку $1+2$ і натисніть <Enter>.

В результаті в командному вікні MATLAB відобразиться наступне:

```
>> 1 + 2
```

```
ans =
```

```
3
```

```
>>
```

Якщо обчислюваний вираз не присвоюється змінній, то створюється змінна *ans* (від слова answer – відповідь). Інформація про змінну *ans* відразу з’явилася у вікні **Workspace** (див. рис. 1.3).

В першому стовпці *Name* записано ім’я змінної.

Наступний стовпець *Value* показує значення змінної у вибраному для її відображення форматі.

Вміст стовпця *Size*, по суті, демонструє основний принцип роботи MATLAB, яка всі дані представляє у вигляді масивів.

Змінна *ans* є двовимірним масивом розміром один на один і займає 8 байт пам’яті, про що свідчить стовпець *Bytes*.

І в останньому стовпці *Class* вказаний тип змінної – *double array*, тобто масив, що складається з чисел подвійної точності.

Будь-який стовпець можна сховати або відобразити, якщо на назві вікна натиснути правою кнопкою мишки і викликати контекстне меню.

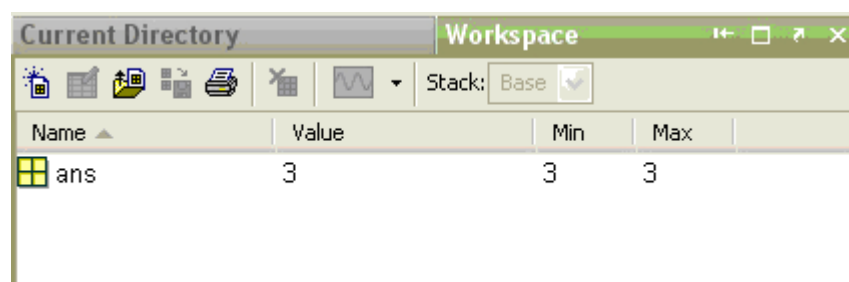


Рисунок 1.3 – Вікно Workspace з інформацією про використання змінних середовища

Якщо необхідно продовжити роботу з попереднім виразом, наприклад, обчислити $(1+2)/4.5$, то найпростіше скористатися існуючим результатом, який зберігається в змінній *ans*. Після набору в командній стрічці *ans/4.5* і натиснення клавіші <Enter>, отримаємо:

```
>> ans/4.5
```

```
ans =
```

```
0.6667
```

```
>>
```

Значення змінної *ans* автоматично замінюється результатом обчислень.

4) Формат виведення результатів обчислень.

Потрібний формат виведення результату визначається користувачем з меню робочого середовища MATLAB.

Виберіть в меню *File* пункт *Preferences*.

На екрані з'явиться діалогове вікно *Preferences*, зображене на рис. 1.4.

В пункті *Command Window* задається формат із списку, що розкривається, *Numeric format* панелі *Text display*.

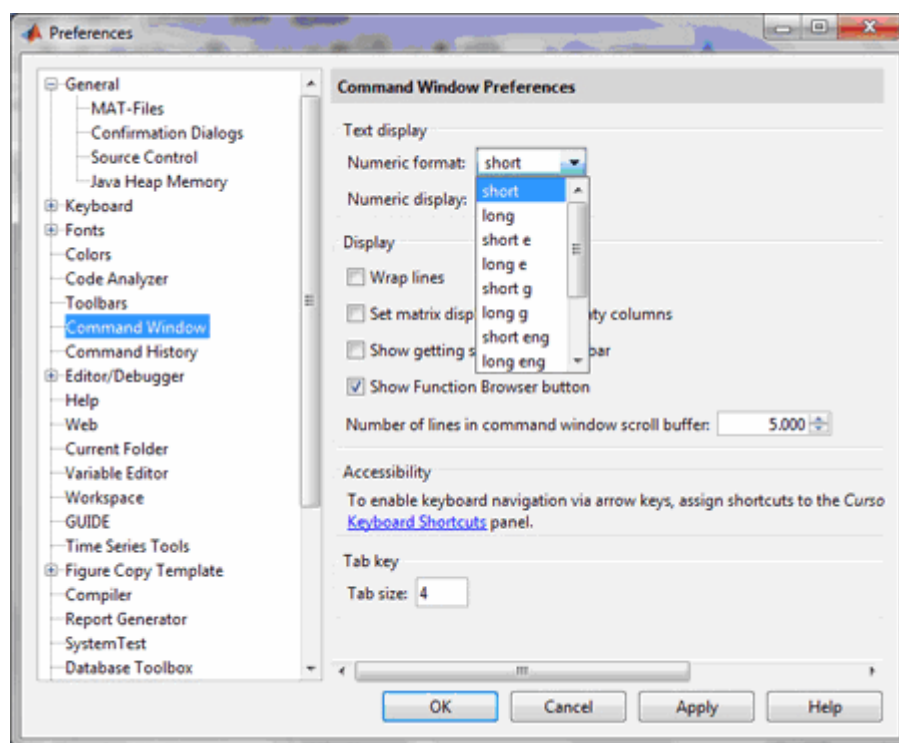


Рисунок 1.4 – Діалогове вікно Preferences

5) Використання елементарних функцій.

Наприклад, необхідно обчислити наступний вираз:

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

Для цього в командній стрічці потрібно ввести даний вираз і натиснути <Enter>:

```
>> exp(-2.5)*log(11.3)^0.3-sqrt((sin(2.45*pi)+cos(3.78*pi))/tan(3.3))
```

Результат виводиться в командне вікно:

```
ans =
```

```
-3.2105
```

Арифметичні операції в MATLAB виконуються в звичайному порядку, властивому для більшості мов програмування.

При проведенні обчислень з комплексними числами в командній стрічці MATLAB можна використовувати i або j , а самі числа при множенні, діленні або піднесенні до степеня необхідно брати у круглі дужки:

```
>> (2.1 + 3.2i)*2 + (4.2 + 1.7i)^2
```

```
ans =
```

```
18.9500 + 20.6800i
```

6) Побудова графіків

Система MATLAB володіє широкими можливостями для графічного представлення результатів обчислень і візуалізації даних.

Виведення відображення функції

$$y(x) = e^{-x} \sin 10x$$

у вигляді графіка складається з наступних етапів:

- задання вектора значень аргумента x ;
- обчислення вектора y значень функції $y(x)$;
- виклик команди *plot* для побудови графіка.

Команди для введення вектора x і обчислення функції краще завершувати крапкою з комою, що дозволяє уникнути виведення в командне вікно їх значень:

```
>> x=0:0.01:1;
```

```
>> y=exp(-x).*sin(10*x);
```

```
>> plot(x,y)
```

Для побудови графіка функції в робочому середовищі MATLAB повинні бути визначені два вектори однакових розмірів, наприклад x та y . Вектор x містить значення аргументів, а y – значення функції цих аргументів.

Команда *plot* з'єднує точки з координатами $(x(i), y(i))$ прямими лініями, автоматично масштабуючи осі для найкращого візуального розміщення графіка по осях.

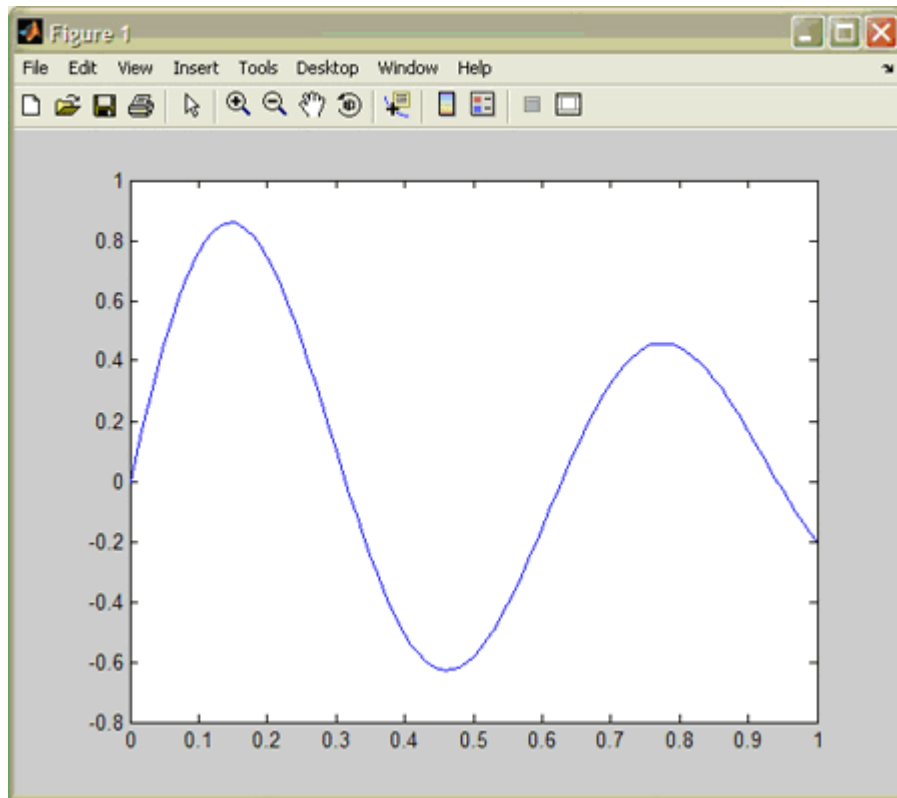


Рисунок 1.5 – Графік функції

Якщо ж відображення графічного результату в системі за замовчуванням користувачу не підходить, можна параметри візуалізації задати індивідуально для кожного графіка, або налаштувати їх вже після відображення результату. Використовуючи кнопку *Dock Figure* (справа в стрічці меню вікна) можна вмонтувати графічне вікно в робоче середовище так, як це показано на рис. 1.6.

Порівняння декількох функцій зручно виконувати, якщо їх графіки відобразити в одній системі координат.

Для побудови графіків функцій $f(x) = \sin(1/x^2)$ і $g(x) = \sin(1.2/x^2)$ потрібно набрати наступну послідовність команд:

```
>> x = -1:0.005:-0.3;
```

```
>> f = sin(x.^-2);
>> g = sin(1.2*x.^-2);
>> plot(x, f, x, g)
```

Також з допомогою команди `plot` можна задати стиль та колір ліній, наприклад:

```
>> plot(x, t, 'k-', x, g, 'k:')
```

Аргументи `'k-'` і `'k:'` задають стиль і колір першої та другої ліній. Тут `k` означає чорний колір, а дефіс або двокрапка – неперервну або пунктирну лінію.

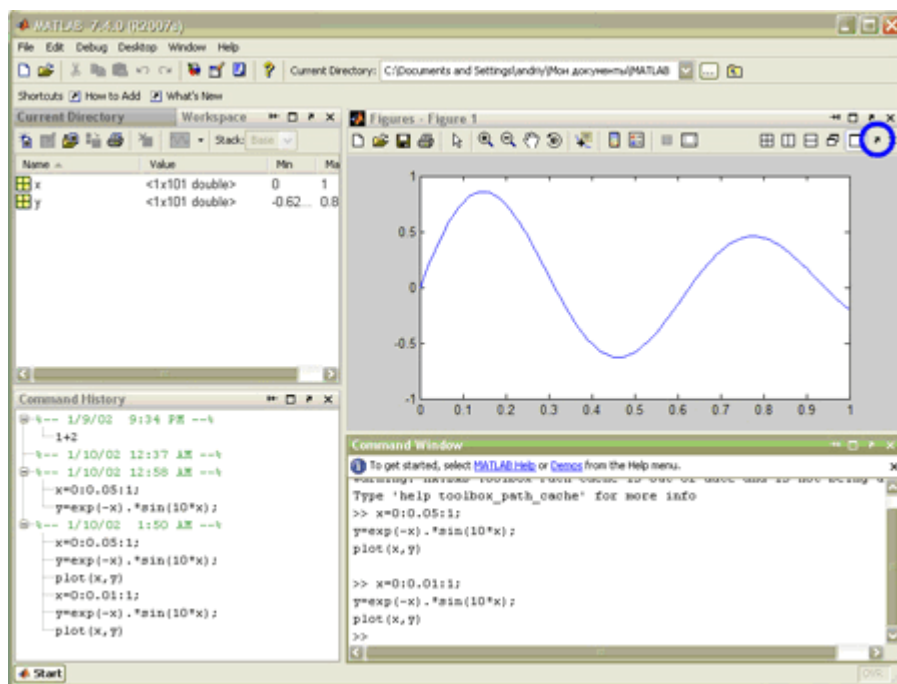


Рисунок 1.6 – Розміщення вікна з графіком у середовищі MATLAB

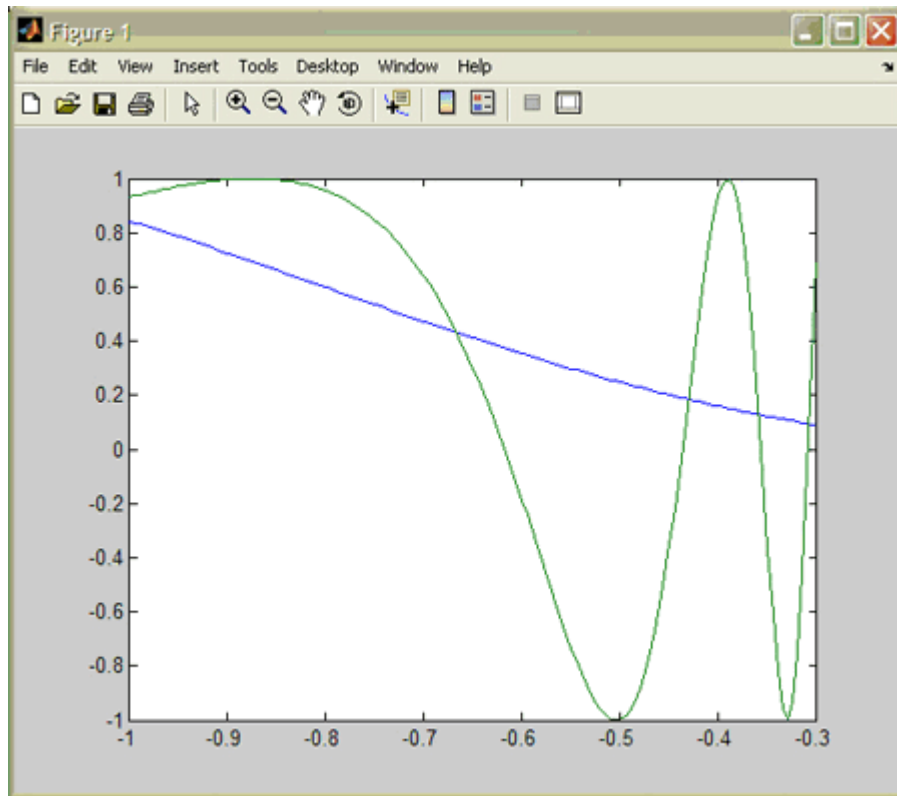


Рисунок 1.7 – Два графіка на одних осях

7) Файли-функції.

Для вирішення обчислювальних задач і написання власних застосунків в MATLAB часто потрібно реалізовувати функції користувача, які здійснюють необхідні дії з вхідними аргументами і повертають результат обчислень. Кількість вхідних і вихідних аргументів залежить від задачі, що вирішується – може бути тільки один вхідний і вихідний аргумент, а може бути декілька вхідних і вихідних, або ж тільки вхідні аргументи. Вхідними аргументами і вихідними даними можуть бути масиви даних довільної розмірності.

Створення програми в середовищі MATLAB здійснюється за допомогою вбудованого редактора. Вікно цього вбудованого редактора виникає на екрані, якщо перед цим використано команду "M-file" із поділу New або обрано назву одного з існуючих M-файлів при виклику команди *Open M-file* із меню File командного вікна. У першому випадку вікно текстового редактора є порожнім (див. рис. 1.8), у другому – у ньому міститься текст викликаного M-файлу. В обох випадках вікно текстового редактора готове для введення нового тексту або коригування існуючого.

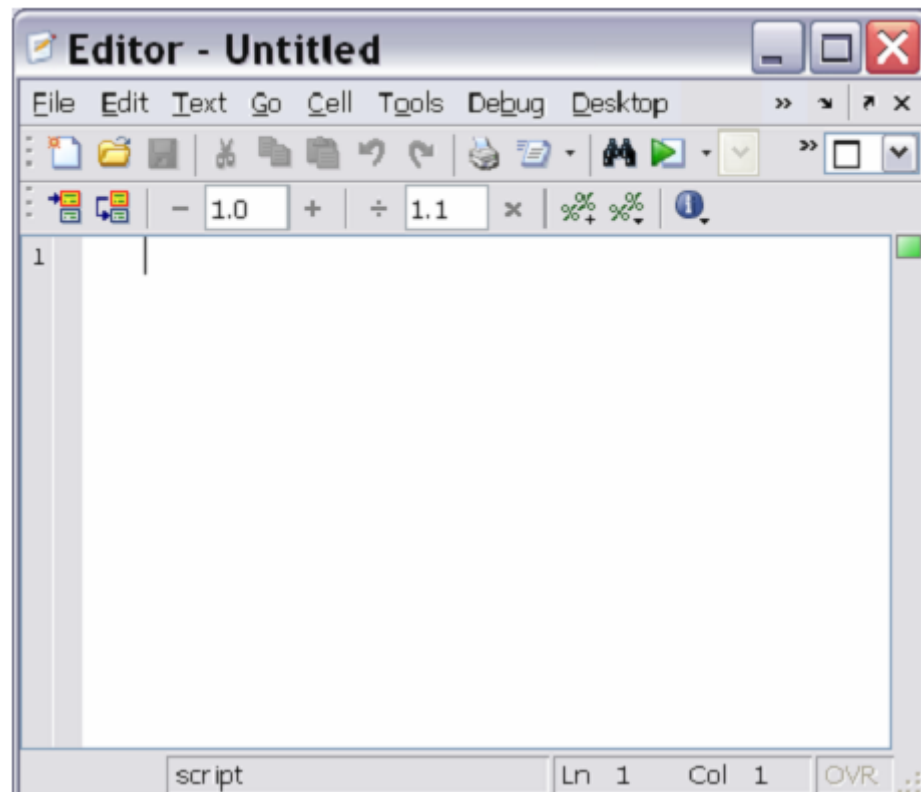


Рисунок 1.8 – Вікно вбудованого текстового редактора Matlab

Запис тексту програми (М-файлу) мовою MATLAB має підпорядковуватися таким правилам:

1) Зазвичай кожний оператор записується в окремому рядку тексту програми. Ознакою кінця оператора є символ (він не виникає у вікні) повернення каретки й переходу на наступний рядок, який вводиться в програму при натисканні клавіші <Enter>, тобто при переході при записі тексту програми на наступний рядок.

2) Можна розміщувати кілька операторів в одному рядку. Тоді попередній оператор у тому ж рядку має закінчуватися символом ' ; ' або ' , '.

3) Можна довгий оператор записувати в декілька рядків. При цьому попередній рядок оператора має завершуватися трьома крапками (' ... ').

4) Якщо черговий оператор не закінчується символом ' ; ', результат його дії при виконанні програми буде виведений у командне вікно. Щоб запобігти виведенню на екран результатів дії оператора програми, запис цього оператора в тексті програми має закінчуватися символом ' ; '.

5) Рядок програми, що починається із символу ' % ', не виконується. Цей рядок сприймається системою Matlab як коментар. Тому для введення коментарю в будь-яке місце тексту програми достатньо почати відповідний рядок із символу ' % '.

6) Рядки коментарю, які передують першому виконуваному (тобто такому, що не є коментарем) оператору програми, сприймаються системою MATLAB як опис програми. Саме ці рядки виводяться в командне вікно, якщо в ньому набрано команду

```
help <ім'я файла>
```

7) У програмах мовою MATLAB відсутній символ закінчення тексту програми.

8) У мові MATLAB змінні не описуються і не оголошуються. Будь-яке нове ім'я, що зустрічається в тексті програми при її виконанні, сприймається системою MATLAB як ім'я матриці. Розмір цієї матриці встановлюється при введенні значень її елементів або визначається діями по встановленню значень її елементів, описаними у попередньому операторі або процедурі. Ця особливість робить мову MATLAB дуже простою у вжитку і привабливою. У мові MATLAB неможливо використання вхідної матриці або змінної, у якій попередньо не введені або обчислені значення її елементів (а значить – і визначені розміри цієї матриці). У протилежному випадку при виконанні програми MATLAB виникне повідомлення про помилку – "Змінна не визначена".

9) Імена змінних можуть містити лише букви латинського алфавіту або цифри і мають починатися з букви. Загальна кількість символів в імені може сягати 30. В іменах змінних можуть використовуватися як великі, так і малі букви. Особливістю мови MATLAB є те, що великі й малі букви в іменах розрізняються системою. Наприклад, символи "a" і "A" можуть використовуватися в одній програмі для позначення різних величин.

Файли-функції з одним аргументом.

Реалізувавши один раз файл-функцію, далі можна її використовувати всюди, де необхідно провести потрібні обчислення для заданого аргументу, але для цього файл повинен бути розміщений в робочому каталозі системи.

Відкрийте в редакторі М-файлів новий файл і наберіть наступне:

```
function f = myfun(x)
f = exp(-x)*sqrt((x^2 + 1)/(x^4 + 0.1));
```

Слово *function* в першій стрічці визначає, що даний файл містить функцію. Перша стрічка є заголовком функції, в якій знаходяться ім'я функції і списки вхідних і вихідних аргументів. Вхідні аргументи записуються в круглих дужках після імені функції. Вихідний аргумент f вказується зліва від знаку «рівне» в заголовку функції. При виборі імені файлу необхідно пам'ятати, що воно повинне співпадати із назвою функції, для якої цей файл реалізовано. Також потрібно потурбуватися про уникнення конфліктів з зайнятими іменами функцій системи MATLAB.

Після заголовку розміщується тіло функції – один або декілька операторів, які реалізують алгоритм отримання значення вихідних змінних з вхідних. Перед запуском файлу потрібно його зберегти в робочому каталозі (*File* → *Save* або *File* → *Save as...*).

Тепер створену функцію можна використовувати так само, як і вмонтовані *sin*, *cos* і інші, наприклад:

```
>> y=myfun(5)
```

```
y =
```

```
0.0014
```

Робота файл-функції з масивом даних

Якщо записати:

```
function f = myfun(x)
f = exp(-x).*sqrt((x.^2 + 1)./(x.^4 + 0.1));
```

то тепер аргументом функції *myfun* може бути як число, так і вектор чи матриця значень, наприклад:

```
>> x=[1.3 7.2];
```

```
>> y=myfun(x)
```

$y =$

0.2600 0.0001

Змінна y , в яку записується результат виклику функції *myfun*, автоматично стає вектором потрібного розміру.

Для того, щоб побудувати графік на відрізку $[0, 4]$ з допомогою файлу-сценарію програми або з командної стрічки необхідно набрати:

```
>> x=0:0.5:4;
```

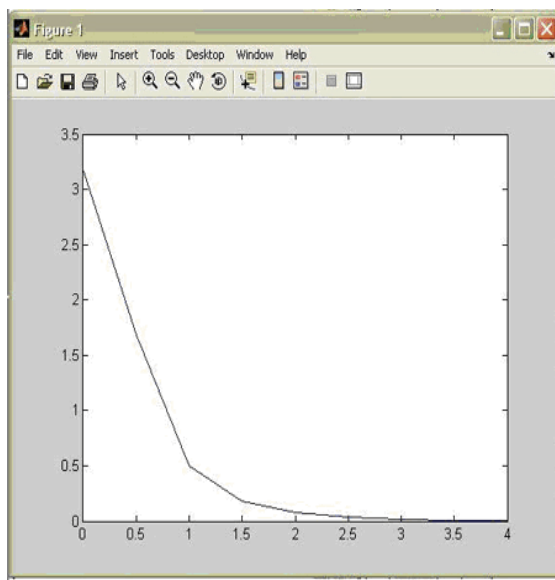
```
>> y=myfun(x);
```

```
>> plot(x,y)
```

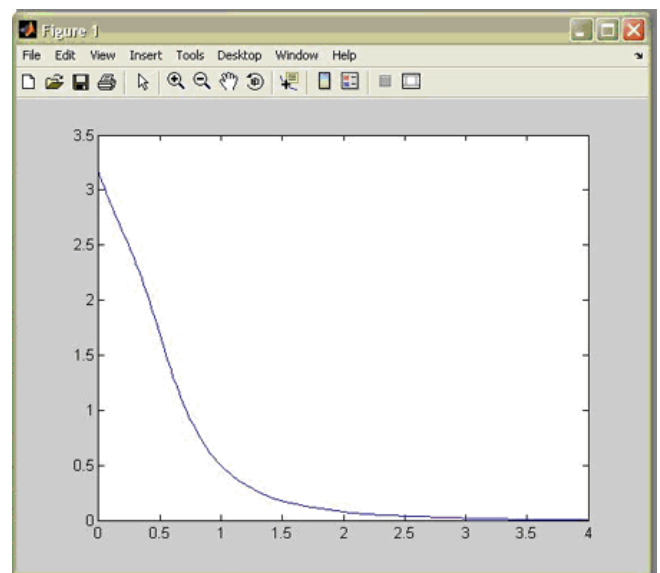
або

```
fplot('myfun',[0 4])
```

Отримаємо результат, зображений на рис. 1.9.



а)



б)

Рисунок 1.9 – Порівняння plot і fplot

Графік побудований з допомогою *fplot* більш точно відображає поведінку функції, тому що алгоритм автоматично підбирає крок аргументу, зменшуючи його на відрізках швидкої зміни досліджуваної функції.

Файл-функція з декількома вхідними аргументами

Написання файлу-функції з декількома вхідними аргументами практично не відрізняється від написання файлу-функції з одним вхідним аргументом.

Нижче наведено приклад файлу-функції з трьома вхідними аргументами для обчислення радіус-вектора точки тривимірного простору:

```
function r=radius3(x,y,z)
r=sqrt(x.^2+y.^2+z.^2);
```

Для обчислення радіус-вектора точки тривимірного простору можна скористатися функцією radius3:

```
>> R=radius3(1,1,1)
R =
1.7321
```

8) Оператори циклу

Схожі і повторювані дії виконуються з допомогою операторів циклу *for* та *while*. Цикл *for* призначений для виконання заданої кількості дій, що повторюються, *while* - для дій, кількість яких наперед невідома, але відома умова виконання і закінчення циклу.

Цикл for

Використання *for* здійснюється наступним чином:

```
for count = start:step:final
команди MATLAB
end
```

Тут *count* – змінна циклу, *start* – її початкове значення, *final* – кінцеве значення, а *step* – крок, на який збільшується *count* при кожному наступному заході у цикл.

Цикл закінчується, як тільки значення *count* стає більше ніж *final*. Змінна циклу може приймати не тільки цілі числа, а й дійсні значення з будь-яким знаком. Наприклад, для виводу графіків кривих для $x \in [0, 2\pi]$, яке задано функцією $y = (x, a) = e^{-ax} \sin(x)$, що залежить від параметра *a*, для значень параметра *a* від -0,1 до 0,1 з кроком 0,002. В редакторі М-файлів потрібно набрати код:

```
figure %
x = 0:pi/30:2*pi; %
for a = -0.1:0.02:0.1
y = exp(-a*x).*sin(x); %
hold on;
```

```
plot(x, y);  
end
```

В результаті виконання даного фрагменту коду з'явиться графічне вікно, яке зображене на рис. 1.10.

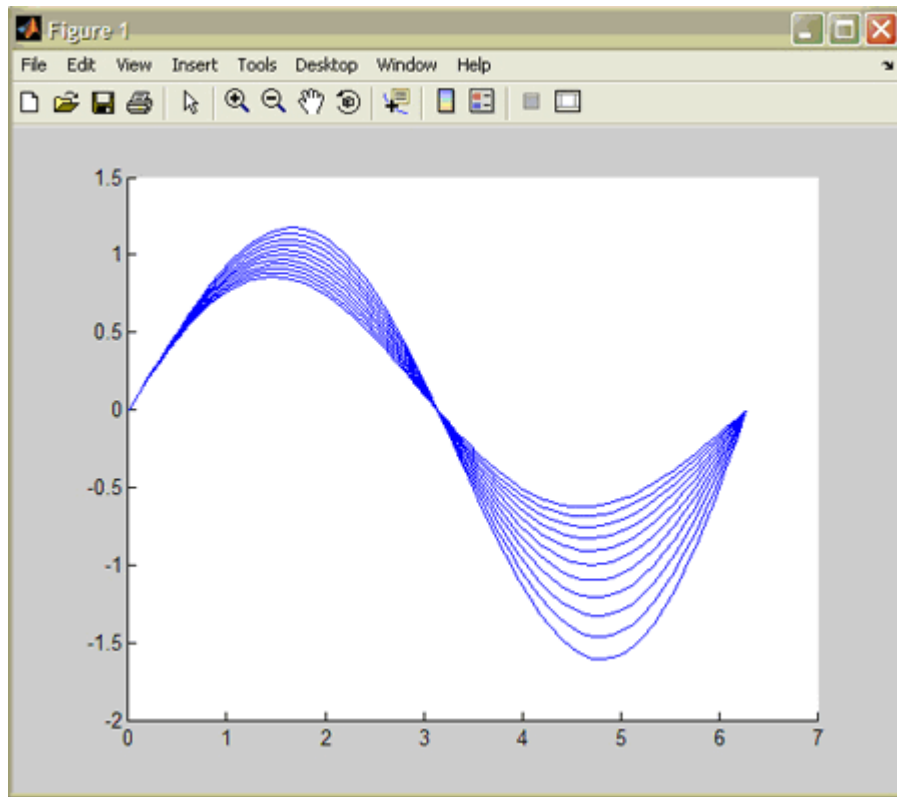


Рисунок 1.10 – Сімейство кривих

Цикл while

Використання *while* здійснюється наступним чином:

```
while (умова повторення циклу)  
команди MATLAB  
end
```

Команди системи MATLAB® виконуються, поки виконується умова повторення циклу.

Умовний оператор if

Умовний оператор **if** в загальному вигляді записується таким чином:

```
if Умова1  
Інструкція_1  
else if Умова2  
Інструкція_2  
else  
Інструкція_3
```

end

Наприклад, поки Умова1 повертає логічне значення 1 (тобто «істина»), виконується Інструкція_1, що складає тіло структури *if...else if*.

Оператор *end* вказує на кінець переліку інструкцій. Інструкції в списку розділяються оператором , (кома) або ; (крапка з комою). Якщо Умова не виконується (дає логічне значення 0, «не істина»), то виконуються інструкції, що записані після ключового слова *else*.

Ще одна конструкція:

if Умова

Інструкція_1

else

Інструкція_2

end

Вконується Інструкція_1, якщо виконується Умова, або Інструкція_2 в протилежному випадку.

Умова записується у вигляді:

Вираз_1 Оператор_відношення Вираз_2, в якості
Операторів_відношення використовуються наступні оператори:

Позначення	Операція відношення
==	Дорівнює
<	Менше
>	Більше
<=	Менше-рівне
>=	Більше-рівне
~=	Не дорівнює

Хід роботи

- 1) Опрацювати теоретичний матеріал по літературі та конспекту.
- 2) Згідно свого варіанту (див. табл. 1.1) обчислити значення функцій *a* та *b* при заданих значеннях аргументу, використовуючи файл-функції.
- 3) Згідно свого варіанту (див. табл. 1.1) побудувати графіки функцій *a* та *b* та графіки цих функцій в одній системі координат за допомогою команд *plot* або *fplot*.

4) Захистити роботу.

Звіт повинен містити

- 1) Тему та мету лабораторної роботи.
- 2) Хід рішення завдання (скопіювати з MATLAB).
- 3) Результати розрахунків та графіки.

Таблиця 1.1 – Варіанти завдань для самостійної роботи

1	a) $x - \sin x = 0,25$	$x=1; 3\pi$	b) $x^3 - 3x^2 + 9x - 8 = 0$
2	a) $\operatorname{tg}(0,85x + 0,1) = x^2$	$x=0; \pi/2$	b) $x^3 - 6x - 8 = 0$
3	a) $\sqrt{x} - \cos(0,387x) = 0$	$x=1; 3\pi/4$	b) $x^3 - 3x^2 + 6x + 3 = 0$
4	a) $\operatorname{tg}(0,4x + 0,4) = x^2$	$x=0; \pi$	b) $x^3 - 0,1x^2 + 0,4x - 1,5 = 0$
5	a) $\lg x - \frac{7}{2x+6} = 0$	$x=2; \pi/3$	b) $x^3 - 3x^2 + 9x + 2 = 0$
6	a) $\operatorname{tg}(0,5x + 0,2) = x^2$	$x=2; \pi/8$	b) $x^2 + x - 5 = 0$
7	a) $3x - \cos x - 1 = 0$	$x=0; \pi/6$	b) $x^3 + 0,2x^2 + 0,5x - 1,2 = 0$
8	a) $x + \lg x = 0,5$	$x=1; 3\pi/2$	b) $x^3 + 3x + 1 = 0$
9	a) $\operatorname{tg}(0,5x + 0,1) = x^2$	$x=2; \pi/4$	b) $x^3 + 0,2x^2 + 0,5x - 2 = 0$
10	a) $x^2 + 4 \sin x = 0$	$x=1; \pi$	b) $x^3 - 3x^2 + 12x - 9 = 0$
11	a) $\operatorname{ctg} 1,5x - x^2 = 0$	$x=2; \pi/3$	b) $x^3 - 0,2x^2 + 0,3x - 12 = 0$
12	a) $\operatorname{tg}(0,4x + 0,3) = x^2$	$x=1; \pi/6$	b) $x^3 - 3x^2 + 6x - 2 = 0$
13	a) $x \lg x - 1,2 = 0$	$x=4; \pi/4$	b) $x^3 - 0,1x^2 + 0,4x - 1,5 = 0$
14	a) $1,8x^2 - \sin 10x = 0$	$x=0; 3\pi/2$	b) $x^3 + 3x^2 + 6x - 1 = 0$
15	a) $\operatorname{ctgx} - \frac{x}{4} = 0$	$x=6; 4\pi/6$	b) $x^3 + 0,1x^2 + 0,4x - 1,2 = 0$

Література

1 Чепак, Л.В. Численные методы. Использование MatLab [Текст] /

Л.В. Чепак, А.Г. Масловская. – Благовещенск, 2005 г. – 69 с.

2 Киреев, В.И. Численные методы в системах и задачах [Текст]:

учеб.пособие / В.И. Киреев, В.А. Пантелеев. – М.:Высш.шк., 2008 г. – 480 с.

З Половко, А.М. Matlab для студента [Текст] / А.М. Половко, П.Н.

Бутусов. – Спб.: «БХВ-Петербург», 2005 г. – 317 с.

Розробив: Ланська С.С.

Розглянуто та схвалено

на засіданні циклової комісії

програмної інженерії

Протокол № 4 від 17.11.2021 р.

Голова комісії _____ Ланська С.С.

ЛАБОРАТОРНА РОБОТА № 2

з навчальної дисципліни «Чисельні методи»

Тема Обчислення кореня алгебраїчного або трансцендентного рівняння
методом поділу відрізка навпіл

Мета Вміти знаходити корені алгебраїчного або трансцендентного
рівняння методом поділу відрізка навпіл в системі MATLAB

Час –2 години

Теоретичні відомості

Метод половинного ділення (бісекції, дихотомії) ґрунтується на дуже простій ідеї: послідовний поділ відрізка навпіл і вибір тієї половини, де знаки функції на кінцях протилежні, тобто є розв'язок (див. рис. 2.1).

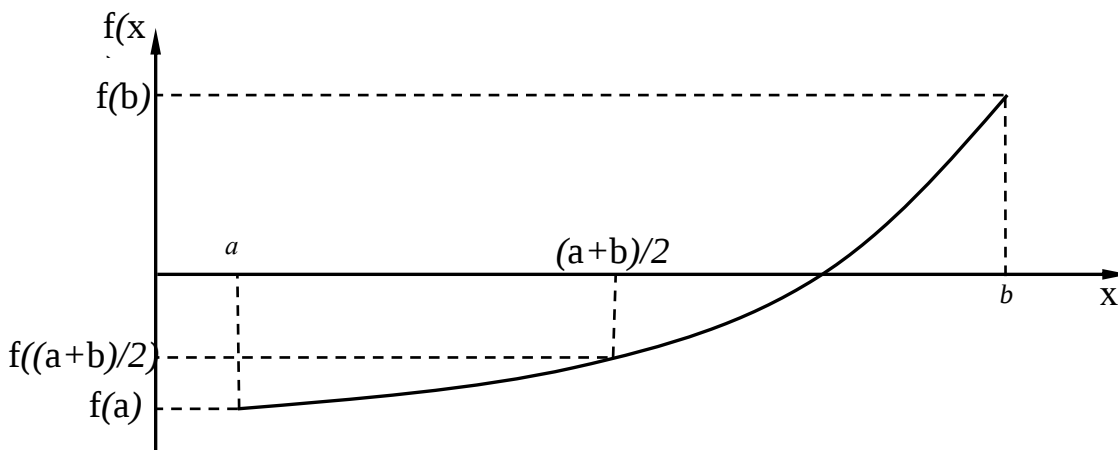


Рисунок 2.1 – Метод половинного ділення

На кінцях початкового відрізка $[a, b]$ знаки функції $f(x)$ різні. З двох половинних відрізків $[a, (a+b)/2]$ та $[(a+b)/2, b]$ треба вибрати другий, бо саме там функція змінює знак. Далі поділ навпіл продовжується на відрізку $[(a+b)/2, b]$. Послідовні наближення припиняються, коли довжина відповідного відрізка стане меншою заданої точності.

Метод простий та надійний, але повільно збігається до розв'язка. Розглянемо рівняння

$$f(x) = 0, \quad (2.1)$$

де $f(x)$ – визначена і безперервна на деякому кінцевому або нескінченному інтервалі.

Всяке значення, що функцію в нуль, називається коренем рівняння (2.1), а спосіб знаходження цього значення і є вирішення рівняння (2.1).

Знайти коріння рівняння виду (2.1) точно вдається лише в окремих випадках. Крім того, часто рівняння містить коефіцієнти, відомі лише приблизно і отже, саме задача про точне визначення коріння рівняння втрачає сенс. Розроблені методи чисельного вирішення рівнянь виду (2.1), що дозволяють відшукати наближені значення коріння цього рівняння.

При цьому доводиться вирішувати два етапи :

1) коріння, тобто відшукування малих областей, в кожній з яких поміщений тільки один корінь рівняння.

2) Обчислення коріння із заданою точністю.

Скористаємося відомим результатом математичного аналізу: якщо безперервна функція приймає на кінцях деякого інтервалу значення різних знаків, то інтервал містить принаймні один корінь рівняння.

Для виділення областей, що містять один корінь, можна використовувати, наприклад, графічний способом, або рухаючись в області визначення деяким кроком, перевіряти на кінцях інтервалів умову зміни знаку функції.

Для вирішення другого етапу існує чисельні методи, один з яких метод половинного ділення.

Провести уточнення коріння методом половинного ділення.

Як початкове наближення виберемо, потім дослідимо функцію на кінцях відрізка. Вибирається той відрізок, у якого значення функції на кінцях має протилежні знаки. Процес поділу відрізка навпіл проводимо до тих пір, поки на якомусь n -му етапі або середина відрізка буде коренем рівняння або

буде отриманий відрізок $[a_n, b_n]$ такий, що $b_n - a_n = (b - a) / 2^n \leq \varepsilon$ і $a_n \leq \xi \leq b_n$ (число n вказує на кількість проведених поділів).

Алгоритм методу:

1) На відрізку $[a, b]$ вибираємо точку x_0 , яка розділяє його на два рівних відрізки $[a, x_0]$ і $[x_0, b]$, довжина яких рівна і знаходиться за формулою $x_0 = \frac{a+b}{2}$.

2) Перевіряємо чи $f(x_0) = 0$, якщо так, то x_0 – точний корінь початкового рівняння і переходимо до пункту 6.

3) У випадку, коли $f(x_0) \neq 0$, то з двох отриманих відрізків $[a, x_0]$ і $[x_0, b]$ вибираємо той, на кінцях якого функція $f(x)$ приймає значення протилежних знаків, тобто, якщо $f(x_0) \cdot f(a) < 0$, тоді залишаємо відрізок $[a, x_0]$ і точку b переносимо в точку x ($b = x_0$); якщо $f(x_0) \cdot f(a) > 0$, то залишаємо відрізок $[x_0, b]$ і переносимо точку a в точку x ($a = x_0$) і переходимо до пункту 1.

4) Процес ділення відрізка навпіл виконується доти, поки на якомусь етапі, або середина відрізка буде коренем, або буде виконана умова закінчення ітераційного процесу: $|b - a| < \varepsilon$.

5) У цьому випадку за наближене значення кореня вибирають $x = \frac{b-a}{2}$.

6) Вивід результатів. Кінець алгоритму.

7) Відомо, що при цьому похибка не перевищує $\frac{b-a}{2^{[k+1]}}$, де k – число ітерацій.

Схема алгоритму розв'язання нелінійного рівняння методом половинного ділення представлена на рисунку 2.2.

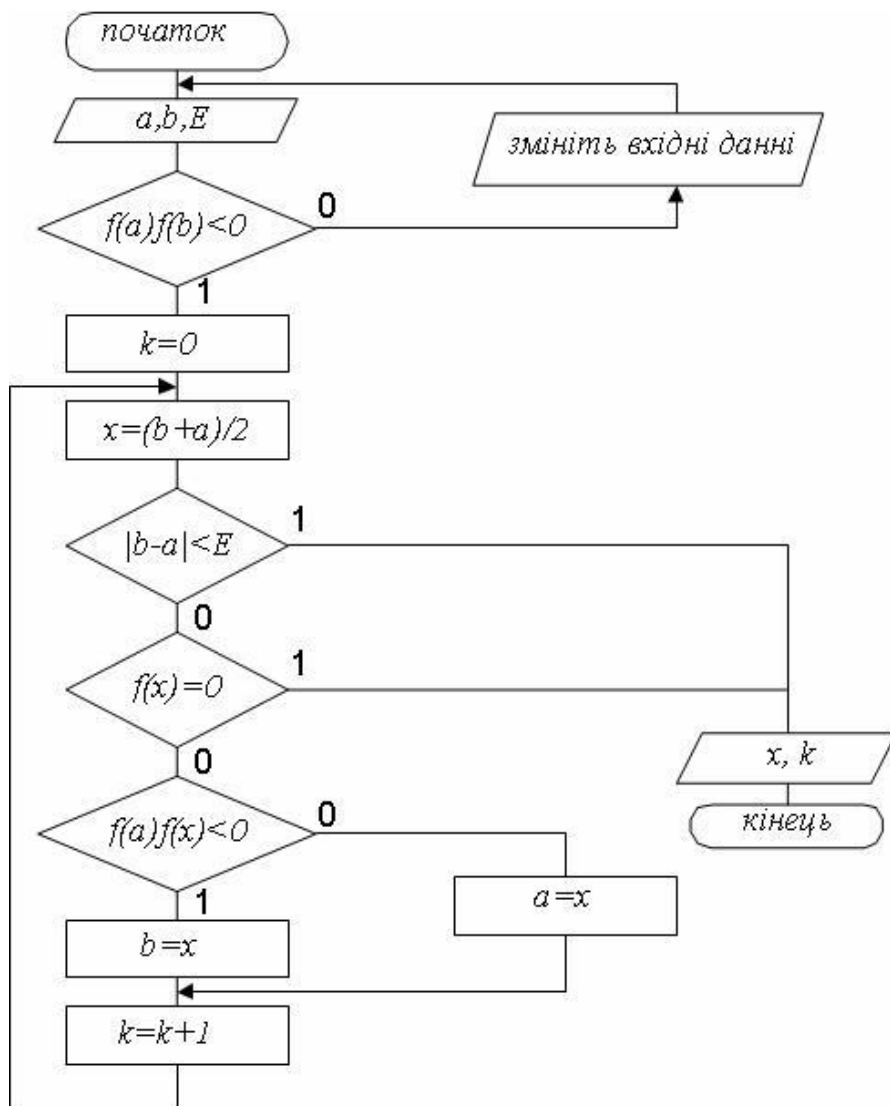


Рисунок 2.2 – Схема алгоритму розв'язання нелінійного рівняння методом половинного ділення

Особливості розробки функцій, які реалізують алгоритм методу:

1) Метод половинного ділення розробляється як незалежні підпрограми-функції з вхідними параметрами: a , b , ε та вихідними: x , k , де x – наближене значення кореня, k – кількість ітерацій.

2) В цих підпрограмах-функціях необхідно передбачити перевірку вхідних даних, наприклад, чи дійсно відрізок вибраний так, що функція на його кінцях має різні знаки.

3) Перед викликом підпрограми-функції, яка реалізує метод, необхідно аналітично визначити кількість коренів. Аналітично чи програмно відокремити корені і в циклі по кількості коренів викликати функцію, яка

реалізує метод уточнення коренів так, щоб на екрані були виведені всі відрізки, корені на кожному з цих відрізків та кількість ітерацій, за яку був отриманий кожен корінь.

Оператори та команди системи MATLAB

Для знаходження коренів полінома в пакеті MATLAB передбачена відповідна функція *roots()*, яка повертає вектор-стовпець, компоненти якого є корінням полінома.

Приклад 1: Знайти корінь рівняння $x^3 + 8x^2 - 14x - 20 = 0$, використовуючи функцію *roots()* (див. рис. 2.3).

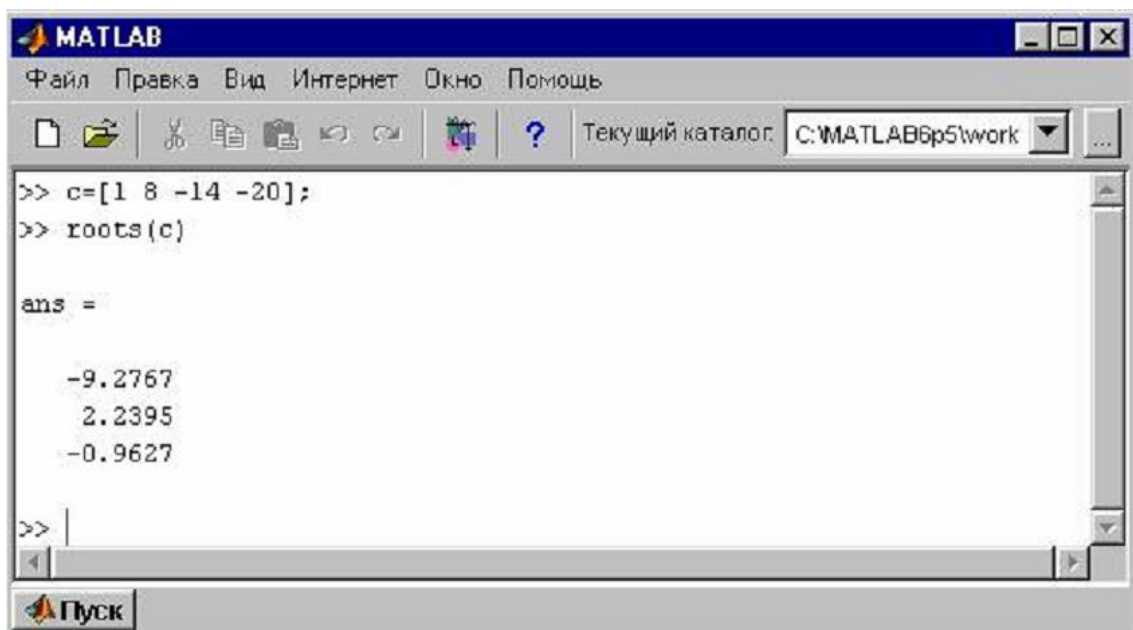


Рисунок 2.3 – Приклад використання функції *roots()*

Функція *fzero* призначена для пошуку кореня рівняння $f(x)=0$. Використовується комбінація методів поділу навпіл, січних і зворотна квадратична інтерполяція. Функція реалізована у вигляді m-файлу. Можливі такі варіанти виклику функції *fzero*:

$$x = fzero(f, x_0)$$

$$x = fzero(f, x_0, options)$$

$$x = fzero(f, x_0, options, P_1, P_2, \dots, P_n)$$

$$[x, fval] = fzero(\dots)$$

$[x, fval, exitflag] = fzero(. . .)$

$[x, fval, exitflag, output] = fzero(. . .)$

Тут f - вказівник на функцію або рядок символів, що містить вираз. В останньому випадку в якості імені аргументу може використовуватися тільки x (ніяка інша буква). Якщо x_0 – скаляр, то $fzero$ знаходить корінь рівняння $f(x)=0$ поблизу x_0 . Якщо $x_0 = [a, b]$ – вектор з двох компонент, то $fzero$ знаходить корінь на відрізку $[a, b]$. В цьому випадку $f(a)$ і $f(b)$ повинні мати різні знаки, інакше видається повідомлення про помилку.

Розглянемо рівняння $\tan x = x$.

Знайдемо найближчий до нуля позитивний корінь рівняння $\tan x = x$:

$[x, fval] = fzero('tan(x) - x', 4)$

Отримаємо $x=4.4934$, $fval=8.8818 \times 10^{-16}$

MATLAB можна використовувати для вирішення алгебраїчних і трансцендентних рівнянь і систем рівнянь, заданих у вигляді масиву символьних виразів. Основний інструмент для цього - функція *solve*. Вона може бути викликана в різній формі:

$solve(E1, E2, ..., EN)$

$solve(E1, E2, ..., EN, var1, var2, ..., varN)$

Тут $E1, E2, ..., EN$ – символьні вирази або змінні, в яких вони містяться, а $var1, ..., varN$ - змінні, відносно яких слід вирішувати систему рівнянь $E1 = 0, E2 = 0, ..., EN = 0$. Зрозуміло, перша форма виклику цієї функції допустима лише у разі, коли немає неоднозначностей щодо того, що саме слід знайти. Функція *solve* повертає єдиний символьний вираз, якщо рівняння (система рівнянь) має єдине рішення і вектор рішень в іншому випадку. Якщо рівняння містить періодичні функції і може тому мати нескінченне число рішень, функція обмежується тим, що повертає коріння за один період в околиці нуля.

Розглянемо приклади.

– єдине рішення:

`>> syms s;`

```
>> E = s + 2;
```

```
>> s = solve(E);
```

```
s =
```

```
-2
```

– кілька рішень:

```
>> syms s;
```

```
>> D = s^2 + 6*s + 9;
```

```
>> s = solve(D)
```

```
s =
```

```
[ -3]
```

```
[ -3]
```

Хід роботи

- 1) Опрацювати теоретичний матеріал по літературі та конспекту.
- 2) Згідно свого варіанту (див. табл. 2.1) визначити області існування коренів рівнянь.
- 3) Розробити алгоритм вирішення рівнянь методом поділу відрізка навпіл за допомогою команд MATLAB.
- 4) Набрати, налагодити та отримати результати рішення, перевірити їх за допомогою однієї з функцій solve (), fzero (), roots ().
- 5) Захистити роботу.

Звіт повинен містити

- 1) Тему та мету лабораторної роботи.
- 2) Постановку задачі та рівняння.
- 3) Основні формули метода та аналітичні розрахунки областей існування коренів рівнянь (скопійовати з MATLAB).
- 4) Результати перевірки рішення.

Завдання для самостійної роботи

Виділити корені трансцендентного та алгебраїчного рівнянь графічно та уточнити їх методом поділу відрізка навпіл з точністю до 0,001.

Таблиця 2.1 – Варіанти завдань для самостійної роботи

1	a) $\operatorname{tg}(0,3x + 0,4) = x^2$	b) $x^3 + 4x - 6 = 0$
2	a) $x^2 - 20 \sin x = 0$	b) $x^3 + 0,2x^2 + 0,5x + 0,8 = 0$
3	a) $\operatorname{ctgx} - \frac{x}{3} = 0$	b) $x^3 - 3x^2 + 12x - 12 = 0$
4	a) $\operatorname{tg}(0,47x + 0,2) = x^2$	b) $x^3 - 0,2x^2 + 0,3x + 1,2 = 0$
5	a) $x^2 + 4 \sin x = 0$	b) $x^3 - 2x + 4 = 0$
6	a) $\operatorname{ctgx} - \frac{x}{2} = 0$	b) $x^3 - 0,2x^2 + 0,5x - 1,4 = 0$
7	a) $2x - \lg x - 7 = 0$	b) $x^3 - 3x^2 + 6x - 5 = 0$
8	a) $\operatorname{tg}(0,44x + 0,3) = x^2$	b) $x^3 - 0,1x^2 + 0,4x + 1,2 = 0$
9	a) $3x - \cos x - 1 = 0$	b) $x^3 - 0,2x^2 + 0,5x - 1 = 0$
10	a) $\operatorname{ctgx} - \frac{x}{10} = 0$	b) $x^3 + 3x^2 + 12x + 3 = 0$
11	a) $x^2 + 4 \sin x = 0$	b) $x^3 - 0,1x^2 + 0,4x + 2 = 0$
12	a) $\operatorname{tg}(0,36x + 0,4) = x^2$	b) $x^3 - 0,2x^2 + 0,4x - 1,4 = 0$
13	a) $x + \lg x = 0,5$	b) $x^3 + 0,4x^2 + 0,6x - 1,6 = 0$
14	a) $\operatorname{ctgx} - \frac{x}{5} = 0$	b) $x^3 + x - 3 = 0$
15	a) $2 \lg x - \frac{x}{2} + 1 = 0$	b) $x^3 - 0,2x^2 + 0,5x + 1,4 = 0$

Література

- 1 Чепак, Л.В. Численные методы. Использование MatLab [Текст] /
Л.В. Чепак, А.Г. Масловская. – Благовещенск, 2005 г. – 69 с.
- 2 Киреев, В.И. Численные методы в системах и задачах [Текст]: учеб.
пособие / В.И. Киреев, В.А. Пантелеев. – М.:Высш.шк., 2008 г. – 480 с.
- 3 Половко, А.М. Matlab для студента [Текст] / А.М. Половко, П.Н. Бутусов.
– Спб.: «БХВ-Петербург», 2005 г. – 317 с.

Розробив: Ланська С.С.

Розглянуто та схвалено

на засіданні циклової комісії

програмної інженерії

Протокол № 4 від 17.11.2021 р.

Голова комісії _____ Ланська С.С.

ЛАБОРАТОРНЕ ЗАНЯТТЯ № 3

з навчальної дисципліни «Чисельні методи»

Тема	Обчислення коренів алгебраїчного або трансцендентного рівняння методом хорд
Мета	Вміти знаходити корені алгебраїчного або трансцендентного рівняння методом хорд в системі MATLAB

Час – 2 години

Теоретичні відомості

У даному методі процес ітерацій полягає в тому, що як наближення до кореню рівняння набувають значень $x_1, x_2, \dots, x_n$ точок перетину хорди AB з віссю абсцис (див. рис. 3.1). Спочатку запишемо рівняння хорди AB :

$$\frac{y - f(a)}{f(b) - f(a)} = \frac{x - a}{b - a}.$$

Для точки перетину хорди AB з віссю абсцис ($x = x_1, y = 0$) отримаємо рівняння:

$$x_1 = a - \frac{f(a)}{f(b) - f(a)}(b - a).$$

Хай для визначеності $f''(x) > 0$ при $a \leq x \leq b$ (випадок $f''(x) < 0$ зводиться до нашого, якщо записати рівняння у вигляді $-f(x) = 0$). Тоді крива $y = f(x)$ буде опукла вниз і, отже, розташована нижче за свою хорду AB . Можливі два випадки:

- 1) $f(a) > 0$ (див. рис. 3.1, а).
- 2) $f(b) < 0$ (див. рис. 3.1, б).

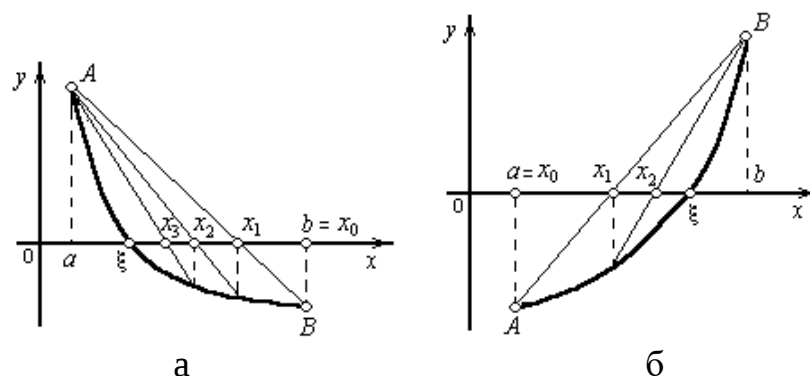


Рисунок 3.1 – Метод хорд

У першому випадку кінець а нерухомий і послідовні наближення: $x_0=b$;

$$x_{i+1} = x_i - \frac{f(x_i)}{f(x_i) - f(a)}(x_i - a), \quad (i = 0, 1, 2, \dots) \quad (3.1)$$

утворюють обмежену монотонну послідовність за зменшенням, причому $a < \xi < \dots < x_{i+1} < x_i < \dots < x_1 < x_0$.

У другому випадку нерухомий кінець b, а послідовні наближення: $x_0=a$;

$$x_{i+1} = x_i - \frac{f(x_i)}{f(b) - f(x_i)}(b - x_i) \quad (3.2)$$

утворюють обмежену монотонно зростаючу послідовність, причому

$$x_0 < x_1 < \dots < x_i < x_{i+1} < \dots < \xi < b.$$

Узагальнюючи ці результати, укладаємо:

– нерухомий той кінець, для якого знак функції $f(x)$ збігається із знаком її другої похідної $f''(x)$;

– послідовні наближення x_n лежать по той бік кореня ξ , де функція $f(x)$ має знак, протилежний знаку її другої похідної $f''(x)$.

Ітераційний процес триває до тих пір, поки не буде виявлено, що

$$|x_i - x_{i-1}| < \varepsilon,$$

де ε – задана гранична абсолютна погрешність.

Алгоритм методу хорд:

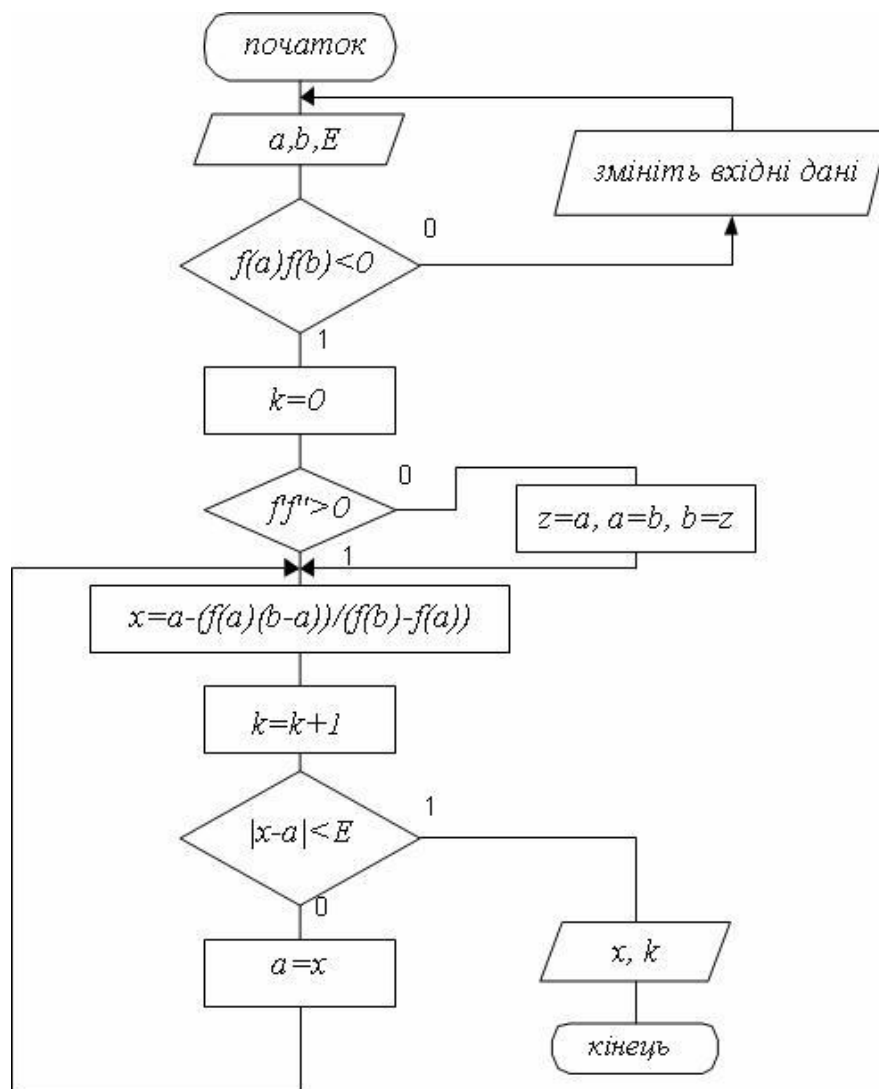


Рисунок 3.1 – Схема алгоритму розв'язання нелінійного рівняння методом хорд

Особливості розробки функцій, які реалізують алгоритм методу:

1) Метод хорд розробляються як незалежні підпрограми-функції з вхідними параметрами: a, b, ξ та вихідними: x, k , де x – наближене значення кореня, k – кількість ітерацій.

2) В цих підпрограмах-функціях необхідно передбачити перевірку вхідних даних, наприклад, чи дійсно відрізок вибраний так, що функція на його кінцях має різні знаки.

3) Обчислення перших та других похідних здійснюється за допомогою спеціальних функцій, в котрих заданий математичний вигляд похідної.

4) Перед викликом підпрограми-функції, яка реалізує метод, необхідно аналітично визначити кількість коренів. Аналітично чи програмно відокремити корені і в циклі по кількості коренів викликати функцію, яка реалізує метод уточнення коренів так, щоб на екрані були виведені всі відрізки, корені на кожному з цих відрізків та кількість ітерацій, за яку був отриманий кожен корінь.

Оператори та команди системи MATLAB

Диференціювання

Щоб знайти похідну символьного виразу, слід скористатися однією з наступних форм функції *diff*:

diff (S) – диференціює вираз S по незалежним змінним, визначеним функцією *findsym*;

diff (S, t) – диференціює вираз S по змінній t;

diff (S, n) – n раз диференціює вираз S

diff (S, t, n) – n раз диференціює вираз S по змінної t;

Наведемо кілька простих прикладів:

```
>> syms s n;
```

```
>> p = s^3 + 4*s^2 - 7*s - 10;
```

```
>> d = diff(p)
```

```
d =
```

```
3*s^2+8*s-7
```

```
>> e = diff(p, 2)
```

```
e =
```

```
6*s+8
```

```
>> g = s^n;
```

```
>> h = diff(g)
```

```
h =
```

```
s^n*n/s
```

```
>> h = simplify(h)
```

```
h =
```

```
s^(n-1)*n
>>
>> f = exp(-(x^2)/2);
>> diff(f)
ans =
-x*exp(-1/2*x^2)
```

Хід роботи

- 1) Опрацювати теоретичний матеріал по літературі та конспекту.
- 2) Згідно свого варіанту (див. табл. 3.1) визначити області існування коренів рівнянь.
- 3) Розробити алгоритм вирішення рівнянь методом хорд за допомогою команд MATLAB.
- 4) Набрати, налагодити та отримати результати рішення, перевірити їх за допомогою однієї з функцій solve (), fzero (), roots ().
- 5) Захистити роботу.

Звіт повинен містити

- 1) Тему та мету лабораторної роботи.
- 2) Постановку задачі та рівняння.
- 3) Основні формули метода та аналітичні розрахунки областей існування коренів рівнянь (скопійовати з MATLAB).
- 4) Результати перевірки рішення.

Завдання для самостійної роботи

Виділити корені трансцендентного та алгебраїчного рівнянь графічно та уточнити їх методом хорд з точністю до 0,001.

Таблиця 3.1 – Варіанти завдань для самостійної роботи

1	a) $x - \sin x = 0,25$	б) $x^3 - 3x^2 + 9x - 8 = 0$.
2	a) $\lg(0,58x + 0,1) = x^2$	б) $x^3 - 6x - 8 = 0$.
3	a) $\sqrt{x} - \cos(0,387x) = 0$	б) $x^3 - 3x^2 + 6x + 3 = 0$.
4	a) $\lg(0,4x + 0,4) = x^2$	б) $x^3 - 0,1x^2 + 0,4x - 1,5 = 0$.
5	a) $\lg x - \frac{7}{2x+6} = 0$	б) $x^3 - 3x^2 + 9x + 2 = 0$.
6	a) $\lg(0,5x + 0,2) = x^2$	б) $x^2 + x - 5 = 0$.
7	a) $3x - \cos x - 1 = 0$	б) $x^3 + 0,2x^2 + 0,5x - 1,2 = 0$.
8	a) $x + \lg x = 0,5$	б) $x^3 + 3x + 1 = 0$.
9	a) $\lg(0,5x + 0,1) = x^2$	б) $x^3 + 0,2x^2 + 0,5x - 2 = 0$.
10	a) $x^2 + 4 \sin x = 0$	б) $x^3 - 3x^2 + 12x - 9 = 0$.
11	a) $\operatorname{ctg} 1,05x - x^2 = 0$	б) $x^3 - 0,2x^2 + 0,3x - 1,2 = 0$.
12	a) $\lg(0,4x + 0,3) = x^2$	б) $x^3 - 3x^2 + 6x - 2 = 0$.
13	a) $x \lg x - 1,2 = 0$	б) $x^3 - 0,1x^2 + 0,4x - 1,5 = 0$.
14	a) $1,8x^2 - \sin 10x = 0$	б) $x^3 + 3x^2 + 6x - 1 = 0$.
15	a) $\operatorname{ctg} x - \frac{x}{4} = 0$	б) $x^3 + 0,1x^2 + 0,4x - 1,2 = 0$

Література

1 Чепак, Л.В. Численные методы. Использование MatLab [Текст] /

Л.В. Чепак, А.Г. Масловская. – Благовещенск, 2005 г. – 69 с.

2 Киреев, В.И. Численные методы в системах и задачах [Текст]: учеб.

пособие / В.И. Киреев, В.А. Пантелеев. – М.: Высш. шк., 2008 г. – 480 с.

3 Половко, А.М. Matlab для студента [Текст] / А.М. Половко, П.Н. Бутусов.

– Спб.: «БХВ-Петербург», 2005 г. – 317 с.

Розробив: Ланська С.С.

Розглянуто та схвалено

на засіданні циклової комісії

програмної інженерії

Протокол № 4 від 17.11.2021 р.

Голова комісії _____ Ланська С.С.

ЛАБОРАТОРНЕ ЗАНЯТТЯ №4
з навчальної дисципліни «Чисельні методи»

Тема Обчислення коренів алгебраїчного або трансцендентного
рівнянь методом Ньютона

Мета Вміти знаходити корені алгебраїчного або трансцендентного
рівняння методом Ньютона в системі MATLAB

Час –2 години

Теоретичні відомості

Відмінність цього ітераційного методу від попереднього полягає в тому, що замість хорди на кожному кроці проводиться дотична до кривій $y=f(x)$ при $x=x_i$ і шукається точка перетину дотичною з віссю абсцис (див. рис. 4.1). При цьому не обов'язково задавати відрізок $[a, b]$, що містить корінь рівняння (4.1), досить знайти лише деяке початкове наближення кореня $x=x_0$.

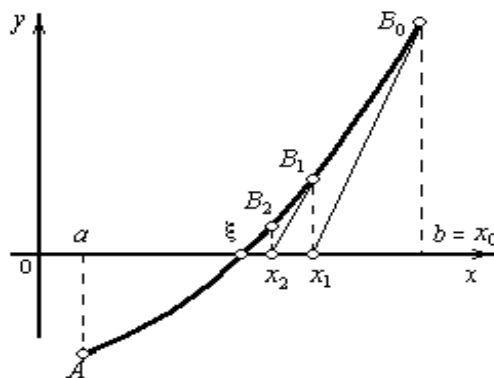


Рисунок 4.1 – Графічна інтерпретація методу дотичних

Застосовуючи метод Ньютона, слід керуватися наступним правилом: як початкова точка x_0 вибирається той кінець інтервалу $[a, b]$, якому відповідає ордината того ж знаку, що і знак $f''(x)$.

Рівняння дотичною, проведеною до кривій $y=f(x)$ через точку B_0 з координатами x_0 і $f(x_0)$, має вигляд:

$$y - f(x_0) = f'(x_0)(x - x_0).$$

Звідси знайдемо наступне наближення кореня x_1 як абсцису точки перетину дотичною з віссю Ox ($y = 0$):

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}.$$

Аналогічно можуть бути знайдені і наступні наближення як точки припинення з віссю абсцис дотичних, проведених в точках B_1, B_2 і так далі. Формула для $i + 1$ наближення має вигляд:

$$x_{i+1} = x_i - \frac{f(x_i)}{f'(x_i)}. \quad (4.1)$$

Для закінчення ітераційного процесу може бути використано або умова $|f(x_i)| < \varepsilon$, або умова близькості 2^x послідовних наближень $|x_i - x_{i-1}| < \varepsilon$.

Ітераційний процес сходиться якщо

$$f(x_0) \cdot f''(x_0) > 0.$$

Алгоритм методу Ньютона (дотичних) представлено на рисунку 4.2

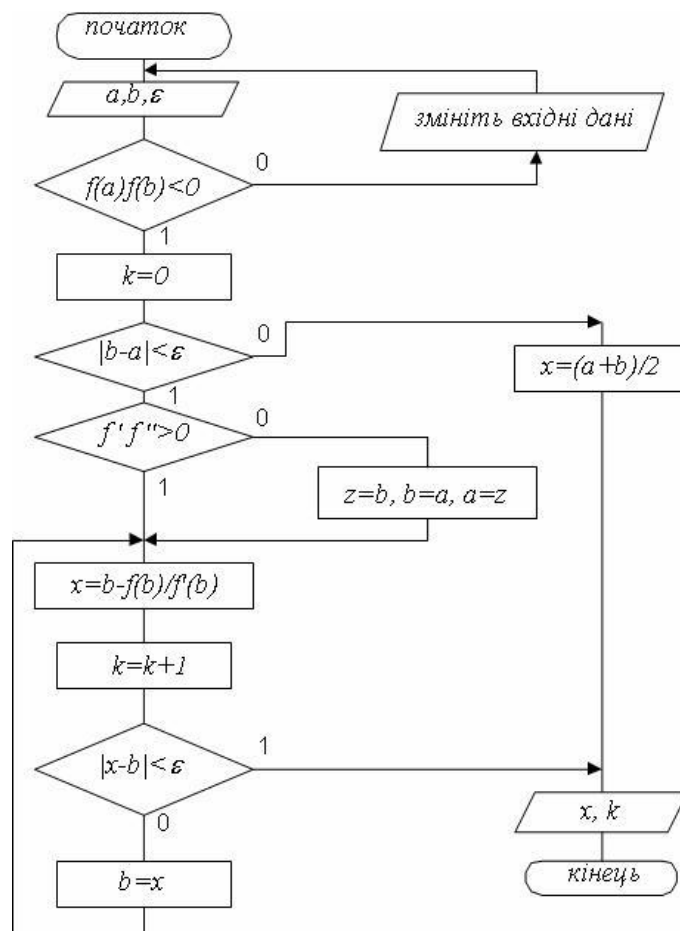


Рисунок 4.2 – Алгоритм розв'язання нелінійного рівняння методом дотичних

Хід роботи

- 1) Опрацювати теоретичний матеріал по літературі та конспекту.
- 2) Згідно свого варіанту (див. табл. 4.1) визначити області існування коренів рівнянь.
- 3) Розробити алгоритм вирішення рівнянь методом Ньютона за допомогою команд MATLAB.
- 4) Набрати, налагодити та отримати результати рішення, перевірити їх за допомогою функцій solve (), fzero (), roots ().
- 5) Захистити роботу.

Звіт повинен містити

- 1) Тему та мету лабораторної роботи.
- 2) Постановку задачі та рівняння.
- 3) Основні формули метода та аналітичні розрахунки областей існування коренів рівнянь (скопювати з MATLAB).
- 4) Результати перевірки рішення.

Завдання для самостійної роботи:

Виділити корені трансцендентного та алгебраїчного рівнянь графічно та уточнити їх методом Ньютона з точністю до 0,001.

Таблиця 4.1 – Варіанти завдань для самостійної роботи

1	a) $tg(0,3x + 0,4) = x^2$	б) $x^3 + 4x - 6 = 0$.
2	a) $x^2 - 20\sin x = 0$	б) $x^3 + 0,2x^2 + 0,5x + 0,8 = 0$.
3	a) $ctgx - \frac{x}{3} = 0$	б) $x^3 - 3x^2 + 12x - 12 = 0$.
4	a) $tg(0,47x + 0,2) = x^2$	б) $x^3 - 0,2x^2 + 0,3x + 1,2 = 0$.
5	a) $x^2 + 4\sin x = 0$	б) $x^3 - 2x + 4 = 0$.
6	a) $ctgx - \frac{x}{2} = 0$	б) $x^3 - 0,2x^2 + 0,5x - 1,4 = 0$.
7	a) $2x - \lg x - 7 = 0$	б) $x^3 - 3x^2 + 6x - 5 = 0$.
8	a) $tg(0,44x + 0,3) = x^2$	б) $x^3 - 0,1x^2 + 0,4x + 1,2 = 0$.
9	a) $3x - \cos x - 1 = 0$	б) $x^3 - 0,2x^2 + 0,5x - 1 = 0$.

Продовження таблиці 4.1

10	a) $\operatorname{ctgx} - \frac{x}{10} = 0$	б) $x^3 + 3x^2 + 12x + 3 = 0.$
11	a) $x^2 + 4\sin x = 0$	б) $x^3 - 0,1x^2 + 0,4x + 2 = 0.$
12	a) $\operatorname{tg}(0,36x + 0,4) = x^2$	б) $x^3 - 0,2x^2 + 0,4x - 1,4 = 0.$
13	a) $x + \lg x = 0,5$	б) $x^3 + 0,4x^2 + 0,6x - 1,6 = 0.$
14	a) $\operatorname{ctgx} - \frac{x}{5} = 0$	б) $x^3 + x - 3 = 0.$
15	a) $2\lg x - \frac{x}{2} + 1 = 0$	б) $x^3 - 0,2x^2 + 0,5x + 1,4 = 0.$

Література

1 Чепак, Л.В. Численные методы. Использование MatLab [Текст] /

Л.В. Чепак, А.Г. Масловская. – Благовещенск, 2005 г. – 69 с.

2 Киреев, В.И. Численные методы в системах и задачах [Текст]:

учеб.пособие / В.И. Киреев, В.А. Пантелеев. – М.:Выш.шк., 2008 г. – 480 с.

3 Половко, А.М. Matlab для студента [Текст] / А.М. Половко, П.Н.

Бутусов. – Спб.: «БХВ-Петербург», 2005 г. – 317 с.

Розробив: Ланська С.С.

Розглянуто та схвалено

на засіданні циклової комісії

програмної інженерії

Протокол № 4 від 17.11.2021 р.

Голова комісії _____ Ланська С.С.

ЛАБОРАТОРНЕ ЗАНЯТТЯ №5

з навчальної дисципліни «Чисельні методи»

Тема	Обчислення коренів алгебраїчного або трансцендентного рівняння методом ітерацій
------	---

Мета	Вміти знаходити корені алгебраїчного або трансцендентного рівняння методом ітерацій в системі MATLAB
------	--

Час –2 години

Теоретичні відомості

Для використання методу ітерації вихідне нелінійне рівняння $f(x) = 0$ заміняється рівносильним рівнянням

$$x = g(x). \quad (5.1)$$

Нехай відомо початкове наближення кореня $x = x_0$. Підставляючи це значення в праву частину рівняння (1), одержимо нове наближення:

$$x_1 = g(x_0).$$

Далі, підставляючи щораз нове значення кореня в (1), одержуємо послідовність значень:

$$x_{i+1} = g(x_i), \quad (i = 0, 1, \dots). \quad (5.2)$$

Геометрично метод ітерації може бути пояснений у такий спосіб. Побудуємо на площині xOy графіки функцій $v = x$ и $v = g(x)$. Кожен дійсний корінь ξ рівняння (5.1) є абсцисою точки перетинання M кривої $v = g(x)$ із прямої $v = x$ (див. рис. 5.1, *a*).

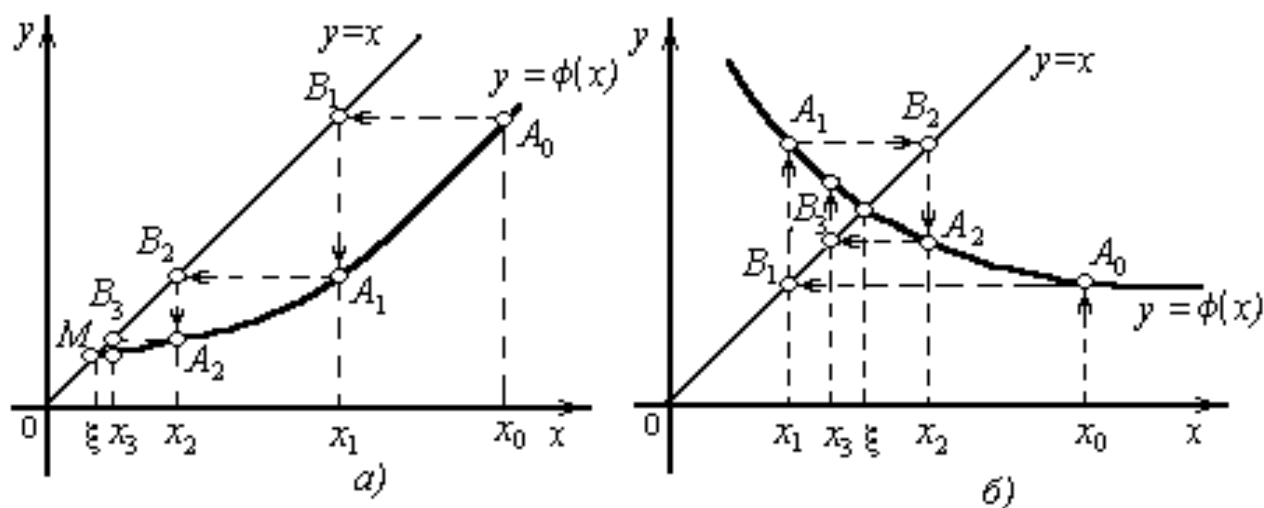


Рисунок 5.1 – Збіжні ітераційні процеси

Відправляючись від деякої точки $A_0 [x_0, g(x_0)]$, будемо ламану $A_0U_1A_1U_2A_2\dots$ (“сходи”), ланки якої поперемінно паралельні осі Ox й осі Oy , вершини $A_0, A_1, A_2, \dots$ лежать на кривій $y = \phi(x)$, а вершини $B_1, B_2, B_3, \dots$, - на прямій $y = x$. Загальні абсциси крапок A_1 й B_1, A_2 й $B_2, \dots$, мабуть, являють собою відповідно послідовні наближення $x_1, x_2, \dots$ корені ξ .

Можливий також інший вид ламаної $A_0U_1A_1U_2A_2 \dots$ – «спіраль» (див. рис. 5.1, б). Рішення у вигляді «сходів» виходить, якщо похідна $\phi'(x)$ позитивна, а рішення у вигляді «спіралі», якщо $\phi'(x)$ негативна.

На рисунку 5.1, а, б крива $y = g(x)$ в околиці кореня ξ - полого, тобто $|\phi'(x)| < 1$, і процес ітерації сходиться. Однак, якщо розглянути випадок, де $|\phi'(x)| > 1$, то процес ітерації може бути розбіжної (див. рис. 5.2). Тому для практичного застосування методу ітерації потрібно з'ясувати достатні умови збіжності ітераційного процесу.

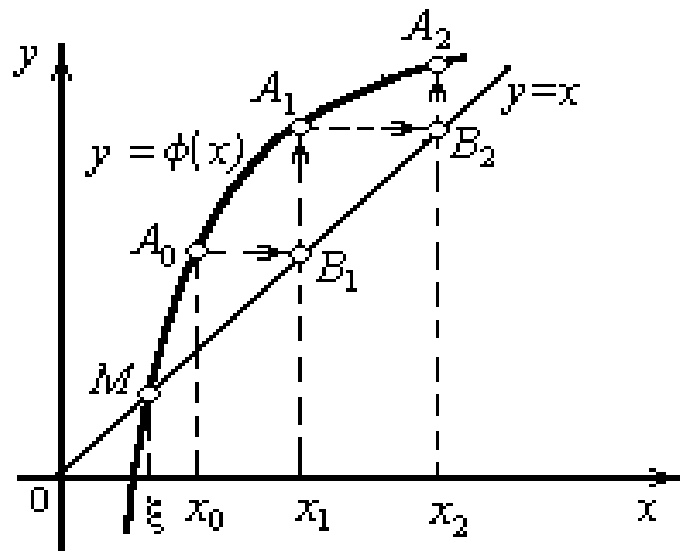


Рисунок 5.2 – Розбіжний ітераційний процес

Теорема: Нехай функція $g(x)$ визначена й диференціюєма на відрізку $[a, b]$, причому всього її значення $g(x) \in [a, b]$.

Тоді, якщо існує правильний дріб q така, що

$$|g'(x)| \leq q < 1$$

при $a < x < b$, то: 1) процес ітерації

$$x_{i+1} = g(x_i), \quad (i = 0, 1, \dots).$$

сходиться незалежно від початкового значення $x_0 \in [a, b]$;

2) граничне значення $\xi = \lim_{n \rightarrow \infty} x_n$ є єдиним коренем рівняння $x = g(x)$ на відрізку $[a, b]$.

Схема алгоритму метода ітерацій представлена на рисунку 5.3.

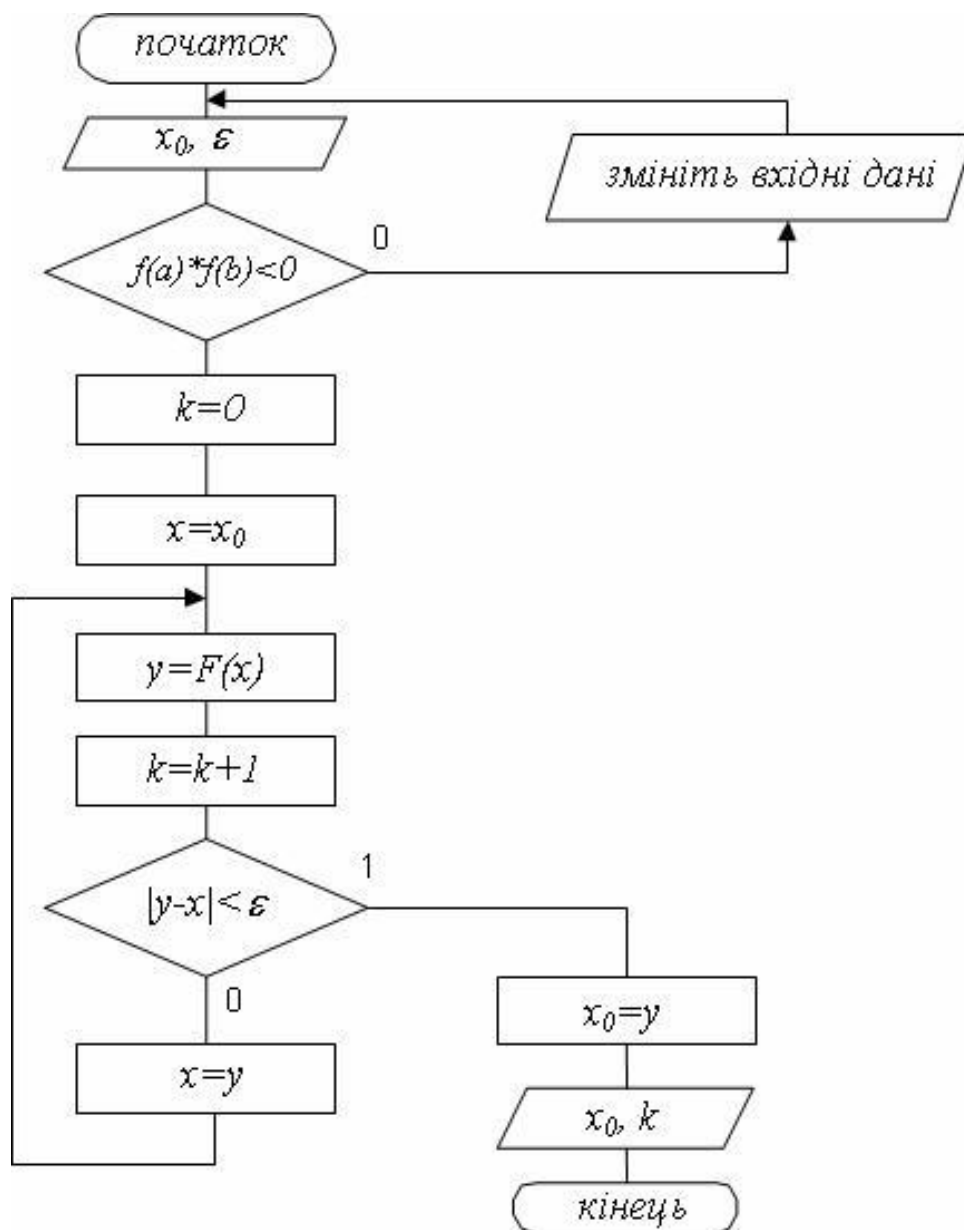


Рисунок 5.3 – Алгоритм розв'язання рівняння методом ітерацій

Хід роботи

- 1) Опрацювати теоретичний матеріал по літературі та конспекту.
- 2) Згідно свого варіанту (див. табл. 5.1) визначити області існування коренів рівнянь.
- 3) Розробити алгоритм вирішення рівнянь методом ітерацій.
- 4) Набрати, налагодити та отримати результати рішення, перевірити їх за допомогою функцій solve (), fzero (), roots ().
- 5) Захистити роботу.

Звіт повинен містити

- 1) Тему та мету лабораторної роботи.
- 2) Постановку задачі та рівняння.
- 3) Основні формули метода та аналітичні розрахунки областей існування коренів рівнянь (скопіювати з MATLAB).
- 4) Результати перевірки рішення.

Завдання для самостійної роботи

Виділити корені трансцендентного та алгебраїчного рівнянь графічно та уточнити їх методом ітерацій з точністю до 0,001.

Таблиця 5.1 - Варіанти завдань для самостійної роботи

1	a) $x - \sin x = 0,25$	б) $x^3 - 3x^2 + 9x - 8 = 0$.
2	a) $\operatorname{tg}(0,58x + 0,1) = x^2$	б) $x^3 - 6x - 8 = 0$.
3	a) $\sqrt{x} - \cos(0,387x) = 0$	б) $x^3 - 3x^2 + 6x + 3 = 0$.
4	a) $\operatorname{tg}(0,4x + 0,4) = x^2$	б) $x^3 - 0,1x^2 + 0,4x - 1,5 = 0$.
5	a) $\lg x - \frac{7}{2x+6} = 0$	б) $x^3 - 3x^2 + 9x + 2 = 0$.
6	a) $\operatorname{tg}(0,5x + 0,2) = x^2$	б) $x^2 + x - 5 = 0$.
7	a) $3x - \cos x - 1 = 0$	б) $x^3 + 0,2x^2 + 0,5x - 1,2 = 0$.
8	a) $x + \lg x = 0,5$	б) $x^3 + 3x + 1 = 0$.
9	a) $\operatorname{tg}(0,5x + 0,1) = x^2$	б) $x^3 + 0,2x^2 + 0,5x - 2 = 0$.
10	a) $x^2 + 4\sin x = 0$	б) $x^3 - 3x^2 + 12x - 9 = 0$.
11	a) $\operatorname{ctg} 1,05x - x^2 = 0$	б) $x^3 - 0,2x^2 + 0,3x - 1,2 = 0$.
12	a) $\operatorname{tg}(0,4x + 0,3) = x^2$	б) $x^3 - 3x^2 + 6x - 2 = 0$.
13	a) $x \lg x - 1,2 = 0$	б) $x^3 - 0,1x^2 + 0,4x - 1,5 = 0$.
14	a) $1,8x^2 - \sin 10x = 0$	б) $x^3 + 3x^2 + 6x - 1 = 0$.
15	a) $\operatorname{ctg} x - \frac{x}{4} = 0$	б) $x^3 + 0,1x^2 + 0,4x - 1,2 = 0$.

Література

- 1 Чепак, Л.В. Численные методы. Использование MatLab [Текст] /
Л.В. Чепак, А.Г. Масловская. – Благовещенск, 2005 г. – 69 с.
- 2 Киреев, В.И. Численные методы в системах и задачах [Текст]:
учеб.пособие / В.И. Киреев, В.А. Пантелеев. – М.:Высш.шк., 2008 г. – 480 с.
- 3 Половко, А.М. Matlab для студента [Текст] / А.М. Половко, П.Н.
Бутусов. – Спб.: «БХВ-Петербург», 2005 г. – 317 с.

Розробив: Ланська С.С.
Розглянуто та схвалено
на засіданні циклової комісії
програмної інженерії
Протокол № 4 від 17.11.2021 р.
Голова комісії _____ Ланська С.С.

з навчальної дисципліни «Чисельні методи»

Час –2 години

Метод Гауса, його ще називають методом Гаусових виключень, полягає в тому, що систему (6.1).

приводять послідовним виключенням невідомих до еквівалентної системи з трикутною матрицею:

рішення якої знаходять по рекурентних формулах:

У матричному записі це означає, що спочатку (прямий хід методу Гауса) елементарними операціями над рядками приводять розширену матрицю системи до ступінчастого вигляду:

$$A_p = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} & b_1 \\ a_{21} & a_{22} & \dots & a_{2n} & b_2 \\ \dots & \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} & b_n \end{bmatrix} \Rightarrow \begin{bmatrix} 1 & \alpha_{12} & \dots & \alpha_{1n} & \beta_1 \\ 0 & 1 & \dots & \alpha_{2n} & \beta_2 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 1 & \beta_n \end{bmatrix},$$

а потім (зворотний хід методу Гауса) цю ступінчасту матрицю перетворюють так, щоб по-перше в стовпцях вийшла одинична матриця:

$$\begin{bmatrix} 1 & 0 & \dots & 0 & x_1 \\ 0 & 1 & \dots & 0 & x_2 \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 1 & x_n \end{bmatrix}.$$

Останній, $(n + 1)$ стовпець цієї матриці містить рішення системи (6.1).

Приклад 6.1. Розв'язати систему рівнянь методом Гауса.

$$\begin{cases} 5x_1 + x_2 + 2x_3 = 10, \\ 6x_1 + 18x_2 + 6x_3 = 54, \\ 10x_1 + 20x_2 + 40x_3 = 160. \end{cases}$$

Розв'язання в математичному пакеті. Процедuru розв'язання системи лінійних рівнянь методом Гауса можна записати як представлено у лістингу 6.1.

Лістинг 6.1 – Процедuru розв'язання системи лінійних рівнянь методом Гауса

```
MGaussSLE = function(A,b,n)
{
  for(k in 1:(n-1))
  {
    for(j in (k+1):n)
      A[k,j] = A[k,j]/A[k,k]
    b[k] = b[k]/A[k,k]
    for(i in (k+1):n)
    {
      for(j in (k+1):n)
        A[i,j] = A[i,j] - A[i,k]*A[k,j]
      b[i] = b[i] - A[i,k]*b[k]
    }
  }
  b[n] = b[n]/A[n,n]
  for(k in (n-1):1)
  {
    for(j in (k+1):n)
      b[k] = b[k] - A[k,j]*b[j]
  }
  return(b)
}
```

Відправними даними задачі (6.1) є матриця A коефіцієнтів при невідомих, вектор-стовпець b вільних членів і кількість невідомих n :

```
> A = cbind(c(5,6,10), c(1,18,20), c(2,6,40))
> b = c(10,54,160)
> n = 3
> A
      [,1] [,2] [,3]
[1,]    5    1    2
[2,]    6   18    6
[3,]   10   20   40
> b
[1] 10 54 160
> n
[1] 3
```

Результат розв'язання заданої системи лінійних рівнянь з використанням записаної процедури:

```
> x = MGaussSLE(A,b,n)
> x
[1] 0.4444444 1.8666667 2.9555556
```

Таким чином, розв'язком заданої системи лінійних рівнянь є знайдені значення невідомих: $x_1=0.444$, $x_2=1.867$, $x_3=2.956$. Для перевірки отриманих результатів знайдені значення невідомих були підставлені у систему:

```
> y = A*x
> y
      [,1]
[1,]    10
[2,]    54
[3,]   160
```

Оператори та команди системи MATLAB

Початкові значення векторів можна задавати із клавіатури шляхом поелементного введення. Для цього в рядку треба спочатку вказати ім'я вектора, потім поставити знак присвоєння '=', а далі, – відкривальну квадратну дужку, а за нею ввести задані значення елементів вектора, відділяючи їх пропусками або комами. Закінчується рядок записом квадратної дужки, що закриває. Наприклад, запис рядка $V = [1.2 \ -0.3 \ 1.2e-5]$ задає вектор V , що містить три елементи зі значеннями 1.2 , -0.3 і $1.2e-5$ (див. рис. 6.1).

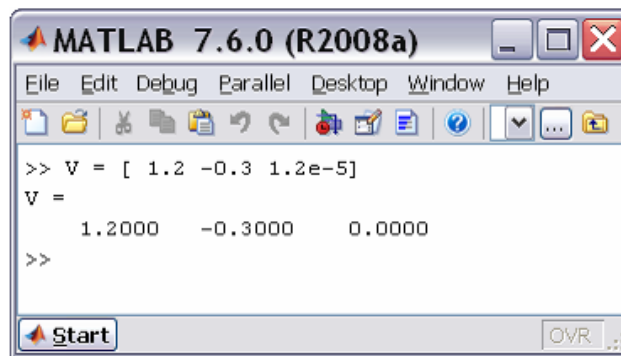


Рисунок 6.1 – Введення вектора

Після введення вектора система виводить його на екран. Те, що в наведеному прикладі останній елемент виведений як 0, обумовлено встановленим форматом *short*, відповідно до якого виводяться дані на екран.

Довгий вектор можна вводити частинами, які потім об'єднують за допомогою операції об'єднання векторів у рядок: $v = [v1 \ v2]$.

Мова MATLAB дає користувачеві можливість скороченого введення вектора, значення елементів якого є арифметичною прогресією. Якщо позначити: *nz* – початкове значення цієї прогресії (значення першого елемента вектора); *kz* – кінцеве значення прогресії (значення останнього елемента вектора); *h* – різницю прогресії (крок), то вектор можна ввести за допомогою короткого запису

$$V = nz : h : kz .$$

Наприклад, введення рядка $V = -0.1 : 0.3 : 1.4$ приведе до результату, показаному на рис. 6.2.

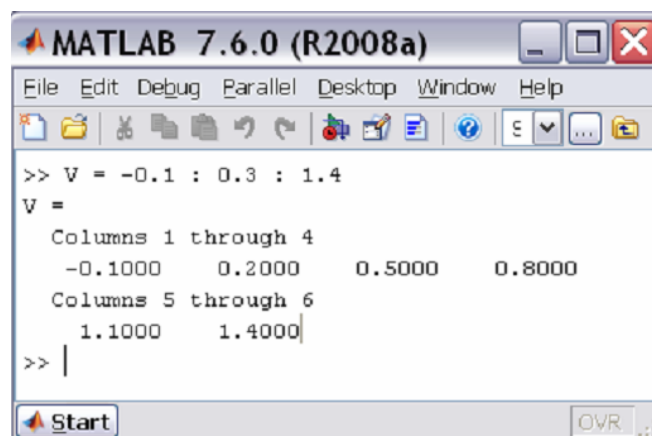


Рисунок 6.2 – Введення вектора арифметичної прогресії

Якщо середній параметр (різниця прогресії) не зазначений, то він за замовчуванням приймається рівним одиниці. Наприклад, команда

```
>> -2:1:5
```

приводить до формування такого вектора

```
ans =  
-2.1000 -1.1000 -0.1000 0.9000 1.9000 2.9000 3.9000 4.9000
```

У такий спосіб вводяться вектори-рядки. Вектор-стовпець вводиться аналогічно, але значення елементів відокремлюються знаком ";".

Введення значень елементів матриці здійснюється в MATLAB у квадратних дужках, по рядках. При цьому елементи рядка матриці один від одного відокремлюються пропуском або комою, а рядки один від одного відокремлюються знаком ";" (див. рис. 6.3).

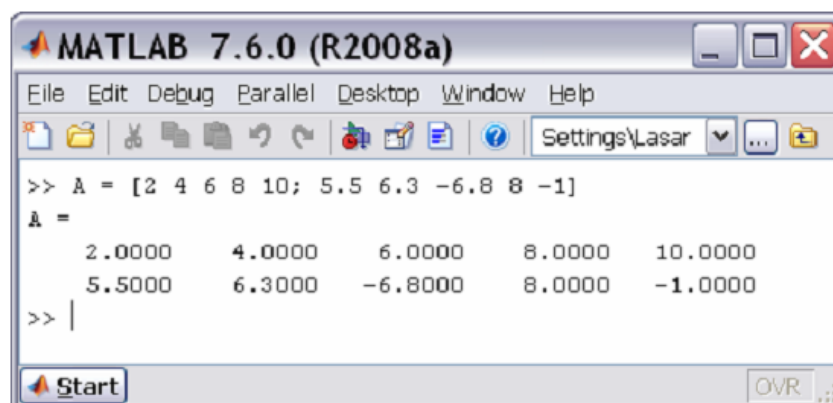


Рисунок 6.3 – Приклад введення матриці

Дії над векторами

Розрізняватимемо дві групи дій над векторами:

- векторні дії – тобто такі, що передбачені векторним з численням у математиці;

- дії по перетворенню елементів – це дії, що перетворюють елементи вектора, але не є операціями, дозволеними математикою з векторами.

Основні матричні операції

У табл. 6.1 наведена інформація про стандартні операції, в яких матричні операнди позначені через x і y, а скалярний операнд - через n.

Таблиця 6.1 – Інформація про стандартні операції над матричними операндами

Формат	Пояснення операції
$X + Y$	Додавання матриць. Операнди і результат мають однакові розміри
$X + n$ або $n + X$	Додаток скаляра до кожного елемента матриці
$X - Y$	Вирахування матриць. Операнди і результат мають однакові розміри
$X - n$ або $n - X$	Віднімання скаляра з кожного елемента матриці або віднімання кожного елемента матриці з скаляра
$X * Y$	Множення матриць. Якщо розмірність X дорівнює $p \times q$, а розмірність Y дорівнює $q \times r$, то розмірність результату - $p \times r$
$X * n$ або $n * X$	Множення кожного елемента матриці на скаляр
X'	Транспонування матриці. Для речових матриць рядки стають стовпцями, а стовпці - рядками. Для комплексних матриць після цього всі елементи замінюються комплексно-сполученими числами
$X. '$	Транспонування матриці. Комплексні матриці транспонуються без зміни їх елементів
$inv(X)$	Звертання квадратної матриці (обчислення X^{-1})
$X \setminus Y$	Лівий поділ матриць - аналог операції $X^{-1} * Y$
X / Y	Правий поділ матриць - аналог операції $X * Y^{-1}$
X^n	Возведення квадратної матриці в ступінь. Якщо n - ціле, то це еквівалентно множенню $x * x * \dots * x$ (n співмножників). Для речових або комплексних значень n ступінь обчислюється як функція від матриці з використанням власних чисел і власних векторів. Результат операції - квадратна матриця
n^X	Возведення скаляра в матричну ступінь (X - квадратна матриця). Результат обчислюється як функція від X з використанням власних чисел і власних векторів. Результат операції - квадратна матриця

Додавання векторів. Як відомо, складатися (підсумовуватися) можуть тільки вектори однакового типу (тобто такі, які обидва є або векторами-рядками, або векторами-стовпцями), що мають однакову довжину (тобто однакову кількість елементів). Якщо X і Y є саме такими векторами, то їхню суму Z можна одержати, увівши команду $Z = X + Y$, наприклад:

» $x = [1 \ 2 \ 3] ; y = [4 \ 5 \ 6];$

» $v = x + y$

$$v = \begin{matrix} 5 & 7 & 9 \end{matrix}$$

Аналогічно за допомогою арифметичного знака "–" здійснюється віднімання векторів, що мають однакову структуру ($Z = X - Y$).

Наприклад:

$$\gg v = x - y$$

$$v = \begin{matrix} -3 & -3 & -3 \end{matrix}$$

Транспонування вектора здійснюється застосуванням знака апострофу, що записується одразу за записом імені вектора, який транспонується.

Наприклад:

$$\gg x'$$

$$ans =$$

$$1$$

$$2$$

$$3$$

Множення вектора на число здійснюється в MATLAB за допомогою знака арифметичного множення ($*$) у такий спосіб: $Z = X*r$ або $Z = r*X$, де r – деяке дійсне число.

Приклад:

$$\gg v = 2*x$$

$$v = \begin{matrix} 2 & 4 & 6 \end{matrix}$$

Множення двох векторів визначено у математиці тільки для векторів однакового розміру (довжини) і лише тоді, коли один із векторів-множників є рядком, а другий – стовпчиком. Тобто, якщо вектори X і Y є рядками, то математичний зміст мають лише дві форми множення цих векторів: $U = X * Y$ і $V = X * Y'$. Причому в першому випадку результатом буде квадратна матриця, а в другому – число.

У MATLAB множення векторів здійснюється застосуванням звичайного знака множення ($*$), який записується між множниками-векторами.

Приклад:

```
» x = [1 2 3] ; y = [ 4 5 6];
```

```
» v = x' * y
```

```
v =
```

```
4    5    6
```

```
8   10   12
```

```
12   15   18
```

```
» v = x * y'
```

```
v =   32
```

У мові MATLAB є ряд операцій, які перетворюють заданий вектор в інший того ж розміру й типу, але в той же час не є математичними операціями з вектором як математичним об'єктом. Усі ці операції перетворюють елементи вектора як елементи звичайного одновимірного масиву чисел. До таких операцій належать, наприклад, усі елементарні математичні функції, наведені раніше і які залежать від одного аргументу.

У мові MATLAB запис, наприклад, виду $Y = \sin(X)$, де X – деякий відомий вектор, приводить до формування нового вектора Y , що має той самий тип і розмір, але елементи якого дорівнюють синусам відповідних елементів вектора-аргументу X .

Наприклад:

```
» x = [-2,-1,0,1,2];
```

```
» y = sin(x)
```

```
y = -0.9093 -0.8415  0  0.8415  0.9093
```

```
» z = tan(x)
```

```
z =  2.1850 -1.5574  0  1.5574 -2.1850
```

```
» v = exp(x)
```

```
v =  0.3679  1.0000  2.7183  7.389
```

Крім цих операцій у MATLAB передбачено декілька операцій поелементного перетворення, що здійснюються за допомогою знаків звичайних арифметичних дій. Ці операції застосовуються до векторів однакового типу й розміру. Результатом їх є вектор того ж типу й розміру.

Додавання (віднімання) числа до (від) кожного елемента вектора. Здійснюється за допомогою знаку "+" ("–").

Поелементне множення векторів. Проводиться за допомогою сукупності знаків ".*", що записується між іменами векторів, які перемножуються. У результаті утворюється вектор, кожний елемент якого є добутком відповідних елементів векторів –"співмножників".

Поелементне ділення векторів. Здійснюється за допомогою сукупності знаків "/". Результат – вектор, кожний елемент якого є часткою від ділення відповідного елемента першого вектора на відповідний елемент другого вектора.

Поелементне ділення векторів в оберненому напрямку. Здійснюється за допомогою сукупності знаків ".\". В результаті отримують вектор, кожний елемент якого є часткою від ділення відповідного елемента другого вектора на відповідний елемент першого вектора.

Поелементне піднесення до степеня. Здійснюється за допомогою сукупності знаків "^". Результат – вектор, кожний елемент якого є відповідним елементом першого вектора, піднесеним до степеня, розмір якого дорівнює значенню відповідного елемента другого вектора.

Приклади

» $x = [1,2,3,4,5]; y = [-2,1,4,0,5];$

» $\text{disp}(x + 2)$

3 4 5 6 7

» $\text{disp}(y - 3)$

-5 -2 1 -3 2

» $\text{disp}(x. *y)$

-2 2 12 0 25

» $\text{disp}(x. /y)$

Warning: Divide by zero

-0. 5000 2. 0000 0. 7500 Inf 1. 0000

» $\text{disp}(x. \backslash y)$

-2. 0000 0. 5000 1. 3333 0 1. 0000

» *disp(x.^y)*

1 2 81 1 3125

У таблиці 6.2 наведені відомості про функції, що обчислюють найбільш уживані характеристики матриць і векторів.

Таблиця 6.2 – Функції, що обчислюють найбільш уживані характеристики матриць і векторів

Формат виклику	Пояснення
<i>det (X)</i>	Визначник або детермінант квадратної матриці
<i>rank (X)</i>	Ранг матриці – число лінійно незалежних рядків (стовпців). Ранг дорівнює найбільшому порядку ненулевого визначника (мінору), отриманого викреслюванням частини рядків і стовпців
<i>trace (X)</i>	Слід матриці – сума значень елементів, розташованих на головній діагоналі
<i>size (X)</i>	Розмір матриці – вектор з двох компонентів, перший з яких дорівнює числу рядків X, а другий - числу стовпців

Функція *solve()* в випадку вирішення системи рівнянь має вигляд:

```
solve ('f1', 'f2', ..., 'fn')
solve ('f1', 'f2', ..., 'fn', x1, x2, ..., xn)
```

де:

- 'f_i' – i-е рівняння системи i=1, 2, 3, ..., n;
- X_i – i-е невідома i=1, 2, 3, ..., n;

Кожне рівняння системи береться в одинарні лапки і відділяється від попереднього комою.

Перед функцією *solve()* необхідно за допомогою функції *syms* визначити символічні змінні.

Технологію рішення системи рівнянь розглянемо на прикладі.

Нехай необхідно вирішити наступну систему рівнянь:

$$\begin{cases} 3x + y - z = 3 \\ -5x + 3y + 4z = 1 \\ x + y + z = 0.5 \end{cases}$$

Програма вирішення системи рівнянь має вигляд:

```
>> syms x y z;
>> Y = solve('3*x+y-z=3', '-5*x+3*y+4*z=1', 'x+y+z=0,5')
```

Після натиснення клавіші <Enter> отримаємо відповідь в наступному вигляді:

```
x : [ 1x1 sym]
y : [ 1x1 sym]
z : [ 1x1 sym]
```

Програма задачу вирішила, але не видало значення змінних x , y , z . Для їх отримання необхідно скористатися командою $Y.k$, де k – ім'я змінної. В нашому випадку рішення має вигляд:

```
>> Y.x
ans =
    -0.10714
>> Y.y
ans =
    1.96428
>> Y.z
ans =
   -1.35714
```

Хід роботи

- 1) Опрацювати теоретичний матеріал по літературі та конспекту.
- 2) Розробити алгоритм вирішення рівняння, згідно з варіантом (див. табл. 6.3), методом Гауса.
- 3) Набрати, налагодити та отримати результати рішення та перевірити їх за допомогою функцій `solve()`.
- 4) Захистити роботу.

Звіт повинен містити

- 1) Тему та мету лабораторної роботи.
- 2) Постановку задачі та систему.
- 3) Основні формули метода та аналітичні розрахунки методу (скопіювати з MATLAB).
- 4) Результати перевірки рішення.

Завдання для самостійної роботи

Таблиця 6.3 – Варіанти завдань для самостійної роботи

№ варіанта	Система лінійних рівнянь	№ варіанта	Система лінійних рівнянь
1	$\begin{cases} 2x_1 + x_2 + 2x_3 + 3x_4 = 8 \\ 3x_1 + 3x_3 = 6 \\ 2x_1 - x_2 + 3x_4 = 4 \\ x_1 + 2x_2 - x_3 + 2x_4 = 4 \end{cases}$	9	$\begin{cases} 2x_1 + x_2 - 5x_3 + x_4 = -4 \\ x_1 - 3x_2 - 6x_4 = -7 \\ 2x_2 - x_3 + 2x_4 = 2 \\ x_1 + 4x_2 - 7x_3 + 6x_4 = -2 \end{cases}$
2	$\begin{cases} x_1 + 2x_2 + 3x_3 + 4x_4 = 22 \\ 2x_1 + 3x_2 + x_3 + 2x_4 = 17 \\ x_1 + x_2 + x_3 - x_4 = 8 \\ x_1 - 2x_3 - 3x_4 = -7 \end{cases}$	10	$\begin{cases} x_1 + 2x_2 + 3x_3 + 4x_4 = 26 \\ 2x_1 + 3x_2 + 4x_3 + x_4 = 34 \\ 3x_1 + 4x_2 + x_3 + 2x_4 = 26 \\ 4x_1 + x_2 + 2x_3 + 3x_4 = 26 \end{cases}$
3	$\begin{cases} 9x_1 + 10x_2 - 7x_3 - x_4 = 23 \\ 7x_1 - x_3 - 5x_4 = 37 \\ 5x_1 - 2x_3 + x_4 = 22 \\ 4x_1 + x_2 + 2x_3 + 3x_4 = 26 \end{cases}$	11	$\begin{cases} 2x_1 - 8x_2 - 3x_3 - 2x_4 = -18 \\ x_1 - 2x_2 + 3x_3 - 2x_4 = 28 \\ x_2 + x_3 + x_4 = 10 \\ 11x_2 + x_3 + 2x_4 = 21 \end{cases}$
4	$\begin{cases} 6x_1 - x_2 + 10x_3 - x_4 = 158 \\ 2x_1 + x_2 + 10x_3 + 7x_4 = 128 \\ 3x_1 - 2x_2 - 2x_3 - x_4 = 7 \\ x_1 - 12x_2 + 2x_3 - x_4 = 17 \end{cases}$	12	$\begin{cases} 2x_1 - x_2 + 4x_3 + x_4 = 66 \\ 2x_2 - 6x_3 + x_4 = -63 \\ 8x_1 - 3x_2 + 6x_3 - 5x_4 = 146 \\ 2x_1 - 7x_2 + 6x_3 - x_4 = 80 \end{cases}$
5	$\begin{cases} x_1 - 2x_2 + 6x_3 + x_4 = 88 \\ 5x_1 + 2x_3 - 3x_4 = 88 \\ 7x_1 - 3x_2 + 7x_3 + 2x_4 = 181 \\ 3x_1 - 7x_2 + 5x_3 + 2x_4 = 99 \end{cases}$	13	$\begin{cases} 2x_1 - 3x_3 - 2x_4 = -16 \\ 2x_1 - x_2 + 13x_3 + 4x_4 = 213 \\ 3x_1 + x_2 + 2x_3 + x_4 = 72 \\ x_1 - 12x_3 - 5x_4 = -159 \end{cases}$
6	$\begin{cases} x_1 - 2x_2 - 8x_4 = -7 \\ x_1 + 4x_2 - 7x_3 + 6x_4 = -8 \\ x_1 + x_2 - 5x_3 + x_4 = -10 \\ 2x_1 - x_2 + 2x_4 = 7 \end{cases}$	14	$\begin{cases} 7x_1 + 7x_2 - 7x_3 - 2x_4 = 5 \\ 3x_1 + 4x_2 + 5x_3 + 8x_4 = 60 \\ 2x_1 + 2x_2 + 2x_3 + x_4 = 27 \\ 2x_1 - 2x_3 - x_4 = -1 \end{cases}$
7	$\begin{cases} 2x_1 + 2x_2 + 6x_3 + x_4 = 15 \\ -x_2 + 2x_3 + x_4 = 18 \\ 4x_1 - 3x_2 + x_3 - 5x_4 = 37 \\ 3x_1 - 5x_2 + x_3 - x_4 = 30 \end{cases}$	15	$\begin{cases} 6x_1 - 9x_2 + 5x_3 + x_4 = 124 \\ 7x_2 - 5x_3 - x_4 = -54 \\ 5x_1 - 5x_2 + 2x_3 + 4x_4 = 83 \\ 3x_1 - 9x_2 + x_3 + 6x_4 = 45 \end{cases}$
8	$\begin{cases} 4x_1 - 5x_2 + 7x_3 + 5x_4 = 165 \\ 2x_1 + x_2 - 3x_3 - x_4 = -15 \\ 9x_1 + 4x_3 - x_4 = 194 \\ x_1 - x_2 - 2x_3 - 3x_4 = -19 \end{cases}$		

Література

- 1 Чепак, Л.В. Численные методы. Использование MatLab [Текст] /
Л.В. Чепак, А.Г. Масловская. – Благовещенск, 2005 г. – 69 с.
- 2 Киреев, В.И. Численные методы в системах и задачах [Текст]:
учеб.пособие / В.И. Киреев, В.А. Пантелеев. – М.:Высш.шк., 2008 г. – 480 с.
- 3 Половко, А.М. Matlab для студента [Текст] / А.М. Половко, П.Н.
Бутусов. – Спб.: «БХВ-Петербург», 2005 г. – 317 с.

Розробив: Ланська С.С.
Розглянуто та схвалено
на засіданні циклової комісії
програмної інженерії
Протокол № 4 від 17.11.2021 р.
Голова комісії _____ Ланська С.С.

ЛАБОРАТОРНЕ ЗАНЯТТЯ № 7
з навчальної дисципліни «Чисельні методи»

Тема Розв'язування системи лінійних алгебраїчних рівнянь за допомогою
формул Крамера

Мета Вміти розв'язувати системи лінійних рівнянь за допомогою формул
Крамера в системі MATLAB

Час – 2 години

Теоретичні відомості

Системою лінійних алгебраїчних рівнянь (СЛАР) називають таку систему m рівнянь з n невідомими:

$$\begin{cases} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n = b_1; \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n = b_2; \\ \vdots \\ a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n = b_m; \end{cases} \quad (7.1)$$

Тут коефіцієнти a_{ij} ; ($i=1, \dots, m$; $j=1, \dots, n$) та праві частини рівнянь b_i ; ($i=1, \dots, m$) є заданими, а x_j , ($j=1, \dots, n$) – невідомі. Розв'язати СЛАР – значить знайти такі x_j , що задовольняють всі рівняння системи (7.1).

Систему (7.1) зручно записувати у матричному вигляді:

$$AX=B, \quad (7.2)$$

$$\text{де } A = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & & & \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{pmatrix}; \quad X = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}; \quad B = \begin{pmatrix} b_1 \\ b_2 \\ \vdots \\ b_m \end{pmatrix}.$$

Кількість рівнянь СЛАР m у загальному випадку не співпадає з кількістю невідомих n . Матриця системи A тоді є прямокутною. Якщо $m=n$, то матриця A є квадратною, а систему (7.2) називають СЛАР n -го порядку.

Одним із методів розв'язування квадратних СЛАР є метод Крамера (відомий швейцарський математик 18-го сторіччя). Розв'язок системи (7.2) за цим методом є відношеннями визначників:

$$x_1 = \frac{D_1}{D}; \quad x_2 = \frac{D_2}{D}; \quad \text{де} \quad D = \det A; \quad D_1 = \det \begin{pmatrix} b_1 & a_{12} \\ b_2 & a_{22} \end{pmatrix};$$

$$D_2 = \det \begin{pmatrix} a_{11} & b_1 \\ a_{21} & b_2 \end{pmatrix}.$$

За методом Крамера можна розв'язувати СЛАР будь-якого порядку n . Однак із зростанням n метод стає дуже громіздким, бо кількість арифметичних операцій для обчислення визначника приблизно рівна $(n+1)!$, а визначників є $(n+1)$.

Рішення системи методом Крамера

- 1) Вводимо матрицю A .
- 2) Обчислюємо D визначник матриці A .
- 3) Задамо матрицю DX_1 , заміною першого стовпця матриці A , матрицею b . Обчислюємо визначника матриці DX_1 .
- 4) Задамо матрицю DX_2 , заміною другого стовпця матриці A , матрицею b . Обчислюємо визначника матриці DX_2 .
- 5) Задамо матрицю DX_3 , заміною третього стовпця матриці A , матрицею b . Обчислюємо визначника матриці DX_3 .
- 6) Визначаємо рішення системи лінійних рівнянь x_1, x_2, x_3 .

Хід роботи

- 1) Опрацювати теоретичний матеріал по літературі та конспекту.
- 2) Розробити алгоритм вирішення системи, згідно з варіантом (див. табл.7.1), за допомогою формул Крамера.
- 3) Набрати, налагодити, отримати результати рішення та проаналізувати їх.
- 4) Захистити роботу.

Звіт повинен містити

- 1) Тему та мету лабораторної роботи.
- 2) Постановку задачі та систему.
- 3) Основні формули метода та аналітичні розрахунки (скопіювати з MATLAB).

Індивідуальні завдання

Таблиця 7.1 – Варіанти завдань

№ варіанта	Система лінійних рівнянь	№ варіанта	Система лінійних рівнянь
1	$\begin{cases} 2x_1 + x_2 + 2x_3 + 3x_4 = 8 \\ 3x_1 + 3x_3 = 6 \\ 2x_1 - x_2 + 3x_4 = 4 \\ x_1 + 2x_2 - x_3 + 2x_4 = 4 \end{cases}$	9	$\begin{cases} 2x_1 + x_2 - 5x_3 + x_4 = -4 \\ x_1 - 3x_2 - 6x_4 = -7 \\ 2x_2 - x_3 + 2x_4 = 2 \\ x_1 + 4x_2 - 7x_3 + 6x_4 = -2 \end{cases}$
2	$\begin{cases} x_1 + 2x_2 + 3x_3 + 4x_4 = 22 \\ 2x_1 + 3x_2 + x_3 + 2x_4 = 17 \\ x_1 + x_2 + x_3 - x_4 = 8 \\ x_1 - 2x_3 - 3x_4 = -7 \end{cases}$	10	$\begin{cases} x_1 + 2x_2 + 3x_3 + 4x_4 = 26 \\ 2x_1 + 3x_2 + 4x_3 + x_4 = 34 \\ 3x_1 + 4x_2 + x_3 + 2x_4 = 26 \\ 4x_1 + x_2 + 2x_3 + 3x_4 = 26 \end{cases}$
3	$\begin{cases} 9x_1 + 10x_2 - 7x_3 - x_4 = 23 \\ 7x_1 - x_3 - 5x_4 = 37 \\ 5x_1 - 2x_3 + x_4 = 22 \\ 4x_1 + x_2 + 2x_3 + 3x_4 = 26 \end{cases}$	11	$\begin{cases} 2x_1 - 8x_2 - 3x_3 - 2x_4 = -18 \\ x_1 - 2x_2 + 3x_3 - 2x_4 = 28 \\ x_2 + x_3 + x_4 = 10 \\ 11x_2 + x_3 + 2x_4 = 21 \end{cases}$
4	$\begin{cases} 6x_1 - x_2 + 10x_3 - x_4 = 158 \\ 2x_1 + x_2 + 10x_3 + 7x_4 = 128 \\ 3x_1 - 2x_2 - 2x_3 - x_4 = 7 \\ x_1 - 12x_2 + 2x_3 - x_4 = 17 \end{cases}$	12	$\begin{cases} 2x_1 - x_2 + 4x_3 + x_4 = 66 \\ 2x_2 - 6x_3 + x_4 = -63 \\ 8x_1 - 3x_2 + 6x_3 - 5x_4 = 146 \\ 2x_1 - 7x_2 + 6x_3 - x_4 = 80 \end{cases}$
5	$\begin{cases} x_1 - 2x_2 + 6x_3 + x_4 = 88 \\ 5x_1 + 2x_3 - 3x_4 = 88 \\ 7x_1 - 3x_2 + 7x_3 + 2x_4 = 181 \\ 3x_1 - 7x_2 + 5x_3 + 2x_4 = 99 \end{cases}$	13	$\begin{cases} 2x_1 - 3x_3 - 2x_4 = -16 \\ 2x_1 - x_2 + 13x_3 + 4x_4 = 213 \\ 3x_1 + x_2 + 2x_3 + x_4 = 72 \\ x_1 - 12x_3 - 5x_4 = -159 \end{cases}$
6	$\begin{cases} x_1 - 2x_2 - 8x_4 = -7 \\ x_1 + 4x_2 - 7x_3 + 6x_4 = -8 \\ x_1 + x_2 - 5x_3 + x_4 = -10 \\ 2x_1 - x_2 + 2x_4 = 7 \end{cases}$	14	$\begin{cases} 7x_1 + 7x_2 - 7x_3 - 2x_4 = 5 \\ 3x_1 + 4x_2 + 5x_3 + 8x_4 = 60 \\ 2x_1 + 2x_2 + 2x_3 + x_4 = 27 \\ 2x_1 - 2x_3 - x_4 = -1 \end{cases}$
7	$\begin{cases} 2x_1 + 2x_2 + 6x_3 + x_4 = 15 \\ -x_2 + 2x_3 + x_4 = 18 \\ 4x_1 - 3x_2 + x_3 - 5x_4 = 37 \\ 3x_1 - 5x_2 + x_3 - x_4 = 30 \end{cases}$	15	$\begin{cases} 6x_1 - 9x_2 + 5x_3 + x_4 = 124 \\ 7x_2 - 5x_3 - x_4 = -54 \\ 5x_1 - 5x_2 + 2x_3 + 4x_4 = 83 \\ 3x_1 - 9x_2 + x_3 + 6x_4 = 45 \end{cases}$
8	$\begin{cases} 4x_1 - 5x_2 + 7x_3 + 5x_4 = 165 \\ 2x_1 + x_2 - 3x_3 - x_4 = -15 \\ 9x_1 + 4x_3 - x_4 = 194 \\ x_1 - x_2 - 2x_3 - 3x_4 = -19 \end{cases}$		

Література

1 Чепак, Л.В. Численные методы. Использование MatLab [Текст] /

Л.В. Чепак, А.Г. Масловская. – Благовещенск, 2005 г. – 69 с.

2 Киреев, В.И. Численные методы в системах и задачах [Текст]:

учеб.пособие / В.И. Киреев, В.А. Пантелеев. – М.:Высш.шк., 2008 г. – 480 с.

3 Половко, А.М. Matlab для студента [Текст] / А.М. Половко, П.Н.

Бутусов. – Спб.: «БХВ-Петербург», 2005 г. – 317 с.

Розробив: Ланська С.С.

Розглянуто та схвалено

на засіданні циклової комісії

програмної інженерії

Протокол № 4 від 17.11.2021 р.

Голова комісії _____ Ланська С.С.

ЛАБОРАТОРНЕ ЗАНЯТТЯ №8

з навчальної дисципліни «Чисельні методи»

Тема	Інтерполювання функцій за допомогою многочленна Лагранжа
Мета	Вміти інтерполювати функції за допомогою многочленна Лагранжа в системі MATLAB

Час –2 години

Теоретичні відомості

Виходячи з однозначності інтерполяційного многочлена $\varphi(x)$, можна побудувати поліном, коефіцієнти якого визначаються із системи. Позначимо задані значення $f(x_i) = y_i$. Оскільки шуканий поліном $\varphi(x)$ повинен приймати в заданих вузлах $x_0, x_1, \dots, x_n$ значення, що збігаються зі значеннями $f(x_0), f(x_1), \dots, f(x_n)$, можна записати $\varphi(x)$ у вигляді:

$$\varphi(x) = \sum_{j=0}^n y_j \phi_j(x) \quad (8.1)$$

де $\phi_j(x)$ — многочлен степеня n , що у вузлах інтерполяції задовольняє такі умови

$$\phi_j(x_i) = \begin{cases} 1, & \text{якщо } i = j \\ 0, & \text{в інших випадках} \end{cases}$$

Даний варіант запису многочлена $\varphi(x)$ називають інтерполяційним поліномом Лагранжа. Для пошуку $\varphi(x)$ знаходять многочлен степеня n , що перетворюється на нуль у вузлах інтерполяції x_i ($i = 0, 1, \dots, n$) і дорівнює одиниці у точці x_j . Многочлен, що задовольняє ці вимоги, може бути записаний у вигляді:

$$\phi_j(x) = \frac{(x - x_0)(x - x_1) \dots (x - x_{j-1})(x - x_{j+1}) \dots (x - x_n)}{(x_j - x_0)(x_j - x_1) \dots (x_j - x_{j-1})(x_j - x_{j+1}) \dots (x_j - x_n)}$$

Тоді, якщо у виразі (1) $\varphi(x)$ знайдено у вказаний вище спосіб, то інтерполяційний многочлен (1) називається інтерполяційним многочленом Лагранжа. Позначимо його як $L_n(x)$, для того щоб відрізнити цю формулу від інших випадків інтерполяції, остаточно:

$$L_n(x) = \sum_{j=0}^n y_j \frac{(x-x_0)(x-x_1)\dots(x-x_{j-1})(x-x_{j+1})\dots(x-x_n)}{(x_j-x_0)(x_j-x_1)\dots(x_j-x_{j-1})(x_j-x_{j+1})\dots(x_j-x_n)}, \quad (8.2)$$

або у скороченому вигляді

$$g(x) = \sum_{i=0}^n y_i \frac{\prod_{j=0, j \neq i}^n (x - x_j)}{\prod_{j=0, j \neq i}^n (x_i - x_j)} = \sum_{i=0}^n y_i \prod_{j=0, j \neq i}^n \frac{(x - x_j)}{(x_i - x_j)} \quad (8.3)$$

Побудова інтерполяційного многочлена Лагранжа в такому вигляді для кожної конкретної задачі пов'язана зі значними обчислювальними затратами.

Для обчислення полінома (многочлена) Лагранжа не треба попереднього визначення коефіцієнтів полінома шляхом розв'язання системи рівнянь, проте для кожного значення x^* поліном доводиться перераховувати знову. Тобто практичне застосування полінома Лагранжа виправдане лише у випадку, якщо інтерполяційна функція обчислюється у порівняно невеликій кількості значень x . Алгоритм методу Лагранжа показано на рисунку 8.1.

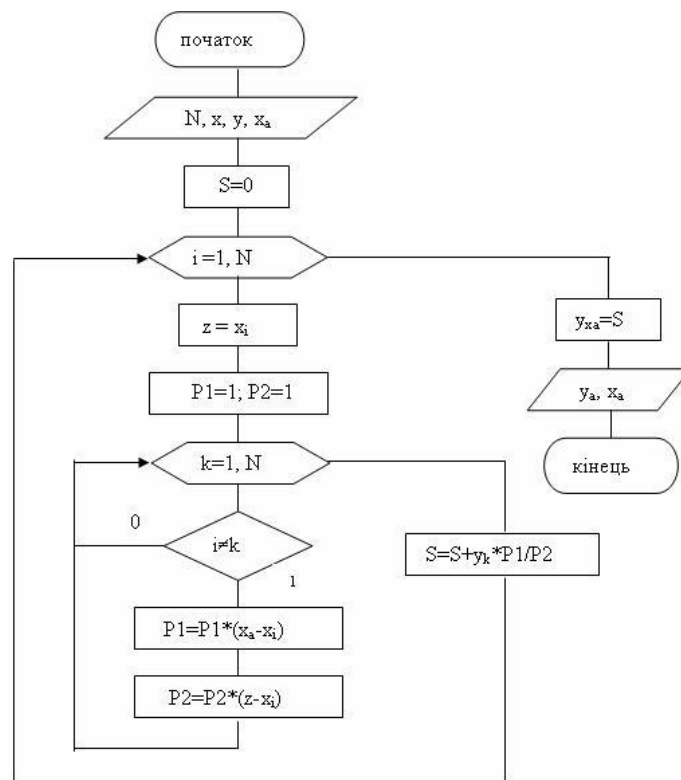


Рисунок 8.1 – Схема алгоритму інтерполяційного многочлена Лагранжа

Хід роботи

- 1) Опрацювати теоретичний матеріал по літературі та конспекту.
- 2) Розробити алгоритми інтерполювання функцій згідно з варіантами (табл. 8.1).
- 3) Набрати, налагодити, отримати результати рішення та їх проаналізувати.
- 4) Захистити роботу.

Звіт повинен містити

- 1) Тему та мету лабораторної роботи.
- 2) Постановку задачі та рівняння.
- 3) Основні формули метода та аналітичні розрахунки (скопіювати з MATLAB).

Завдання для самостійної роботи:

- 1) Згідно заданої таблиці 8.1 значень функції скласти в системі MATLAB формулу інтерполяційного многочлена Лагранжа і побудувати графік.

Таблиця 8.1 – Варіанти завдань для самостійної роботи

№	x_0	x_1	x_2	y_0	y_1	y_2
1	2	3	5	4	1	7
2	4	2	3	5	2	8
3	0	2	3	-1	-4	2
4	7	9	13	2	-2	3
5	-3	-1	3	7	-1	4
6	1	2	4	-3	-7	2
7	-2	-1	2	4	9	1
8	2	4	5	9	-3	6
9	-4	-2	0	2	8	5
10	-1	1.5	3	4	-7	1
11	2	4	7	-1	-6	3
12	-9	-7	-4	3	-3	4
13	0	1	4	7	-1	8
14	8	5	0	9	2	4
15	-7	-5	-4	4	-4	5

2) Обчислити в системі MATLAB одне значення заданої функції для проміжного значення аргументу (а) за допомогою інтерполяційного многочлена Лагранжа. Для виконання завдання початкові данні беруться з таблиць 8.2, 8.3, 8.4 або 8.5.

Таблиця 8.2 – Початкові дані

№ варіанта	Значение а	№ таблиці
1	-2	8.3
2	3.77	8.4
3	0.17	8.5
4	4.83	8.3
5	3.5	8.4
6	0.21	8.5
8	1.75	8.3
8	4.2	8.4
9	0.12	8.5
10	6.76	8.3
11	2.3	8.4
12	0.21	8.5
13	-2.5	8.3
14	5.4	8.4
15	0.33	8.5

Таблиця 8.3 – Значення функції

x	-3.2	-0.8	0.4	2.8	4.0	6.4	7.6
$f(x)$	-1.94	-0.61	0.31	1.81	2.09	1.47	0.68

Таблиця 8.4 – Значення функції

x	1.3	2.1	3.7	4.5	6.1	7.7	8.5
$f(x)$	1.777	4.563	13.84	20.39	37.34	59.41	72.4

Таблиця 8.5 – Значення функції

x	0.10	0.15	0.20	0.25	0.30	0.35	0.40
$f(x)$	0.995	0.988	0.980	0.969	0.955	0.939	0.921

Література

1 Чепак, Л.В. Численные методы. Использование MatLab [Текст] /

Л.В. Чепак, А.Г. Масловская. – Благовещенск, 2005 г. – 69 с.

2 Киреев, В.И. Численные методы в системах и задачах [Текст]:

учеб.пособие / В.И. Киреев, В.А. Пантелеев. – М.:Высш.шк., 2008 г. – 480 с.

3 Половко, А.М. Matlab для студента [Текст] / А.М. Половко, П.Н.

Бутусов. – Спб.: «БХВ-Петербург», 2005 г. – 317 с.

Розробив: Ланська С.С.

Розглянуто та схвалено

на засіданні циклової комісії

програмної інженерії

Протокол № 4 від 17.11.2021 р.

Голова комісії _____ Ланська С.С.

ЛАБОРАТОРНЕ ЗАНЯТТЯ №9

з навчальної дисципліни «Алгоритми та методи обчислень»

Тема	Інтерполювання функцій за допомогою многочленна
	Ньютона.
Мета	Вміти інтерполювати функції за допомогою многочленна
	Ньютона в системі MATLAB.

Час –2 години

Теоретичні відомості

Обчислення значень функції для значень аргументу, що знаходяться на початку таблиці, зручно проводити, користуючись першою інтерполяційною формулою Н'ютона.

Нехай функція $f(x)$ задана значеннями

$$y_0 = f(x_0), \quad y_1 = f(x_1), \quad \dots, \quad y_n = f(x_n) \quad (9.1)$$

у рівновіддалених вузлах інтерполяції $x_0, \quad x_1 = x_0 + h, \quad \dots, \quad x_n = x_0 + nh$.

Необхідно побудувати інтерполяційний многочлен $P_n(x)$ степені n такий, що

$$P_n(x_0) = y_0, \quad P_n(x_1) = y_1, \quad \dots, \quad P_n(x_n) = y_n$$

В силу одиничності многочлена степені n , побудованого по $n+1$ значенням функції $f(x)$, многочлен $P_n(x)$ являється різновидом запису інтерполяційного многочлену і у кінцевому рахунку співпадає із многочленом, отриманим по формулі Лагранжа.

Перша інтерполяційна формула Н'ютона (9.2)

$$P_n(x) = y_0 + \frac{\Delta y_0}{1!h}(x - x_0) + \frac{\Delta^2 y_0}{2!h^2}(x - x_0)(x - x_1) + \dots \\ \dots + \frac{\Delta^n y_0}{n!h^n}(x - x_0)\dots(x - x_{n-1}) \quad (9.2)$$

Для інтерполірування у кінці таблиці за звичай застосовують другу інтерполяційну формулу Н'ютона (9.3).

$$\begin{aligned}
 P_n(x) = & y_n + \frac{\Delta y_{n-1}}{1!h}(x-x_n) + \frac{\Delta^2 y_{n-2}}{2!h^2}(x-x_n)(x-x_{n-1}) + \\
 & + \frac{\Delta^3 y_{n-3}}{3!h^3}(x-x_n)(x-x_{n-1})(x-x_{n-2}) + \dots \\
 & \dots + \frac{\Delta^n y_0}{n!h^n}(x-x_n)\dots(x-x_1).
 \end{aligned}
 \tag{9.3}$$

Алгоритм методу Ньютона показано на рис. 9.1.

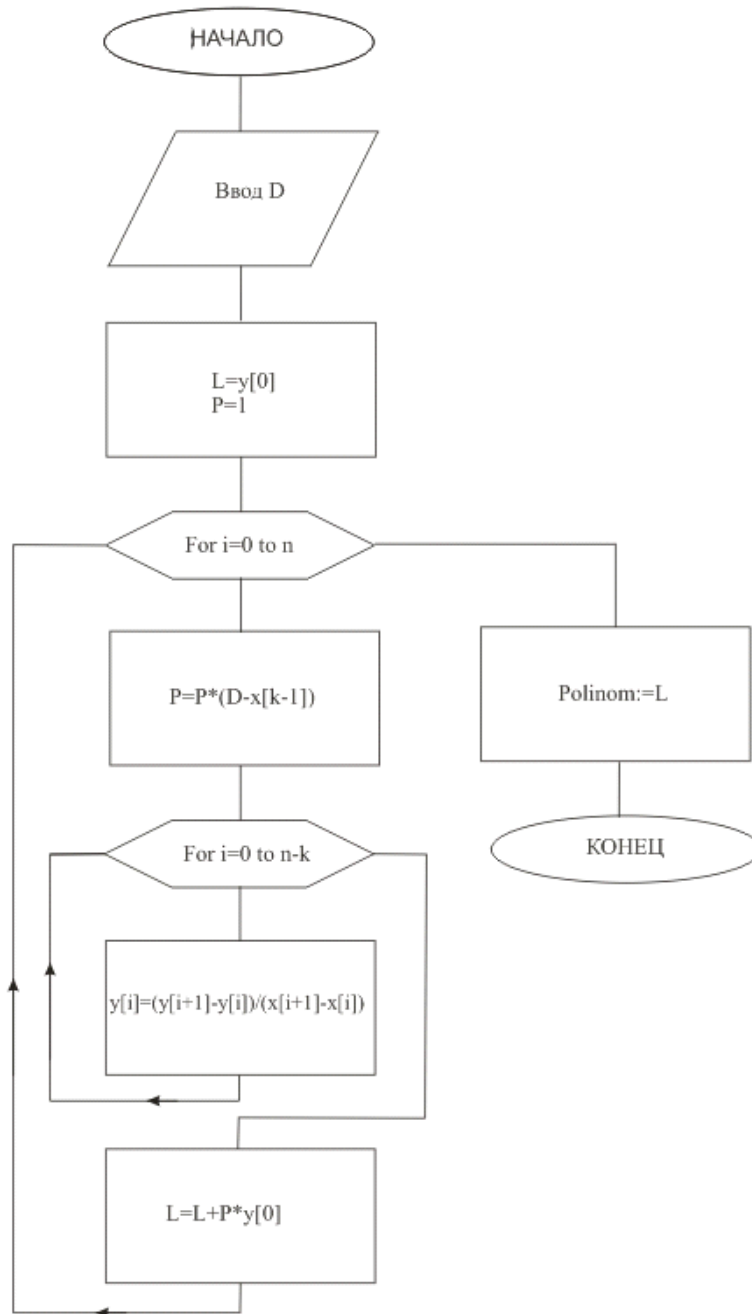


Рисунок 9.1 – Схема алгоритму інтерполяції по Ньютону

MATLAB має функцію `diff`, яка по заданому вектору x будує вектор різниць вказаного порядку:

$dx = \text{diff}(x)$ % обчислення різниці 1-го порядку;

$d^n x = \text{diff}(x, n)$ % обчислення різниці n-го порядку.

В прикладі наводиться фрагмент програми, результати роботи якої для більшої наочності перенесені до таблиці 9.1.

Наприклад:

» $x = 1:0.1:2$;

» $dx = \text{diff}(x)$;

» $y = \log(x)$;

» $dy = \text{diff}(y)$;

» $d2y = \text{diff}(y, 2)$;

Таблиця 9.1 – Результати обчислення кінцевих різниць

x	dx	y	dy	d2y
1.0	0.1	0	0.0953	-0.0083
1.1	0.1	0.0953	0.0870	-0.0070
1.2	0.1	0.1823	0.0800	-0.0059
1.3	0.1	0.2624	0.0741	-0.0051
1.4	0.1	0.3365	0.0690	-0.0045
1.5	0.1	0.4055	0.0645	-0.0039
1.6	0.1	0.4700	0.0606	-0.0035
1.7	0.1	0.5306	0.0572	-0.0031
1.8	0.1	0.5878	0.0541	-0.0028
1.9	0.1	0.6419	0.0513	
2.0		0.6931		

Хід роботи

- 1) Опрацювати теоретичний матеріал по літературі та конспекту.
- 2) Розробити алгоритм інтерполювання функції згідно з варіантом (табл. 9.1).
- 3) Набрати, налагодити, отримати результати рішення та їх проаналізувати.
- 4) Захистити роботу.

Звіт повинен містити

- 1) Тему та мету лабораторної роботи.
- 2) Постановку задачі та рівняння.
- 3) Основні формули метода та аналітичні розрахунки (скопійовати з MATLAB).

Завдання для самостійної роботи

Згідно заданої таблиці значень функції скласти в системі MATLAB формулу інтерполяційного многочленна Ньютона, визначити значення функції в точці (t).

Таблиця 9.1 – Варіанти завдань

№	t	№ таблиці
1	0,39	9.2
2	0,93	9.3
3	0,19	9.2
4	0,68	9.3
5	0,12	9.2
6	0,87	9.3
7	0,22	9.2
8	0,72	9.3
9	0,32	9.2
10	0,81	9.3
11	0,36	9.2
12	0,91	9.3
13	0,14	9.2
14	0,66	9.3
15	0,26	9.2

Таблиця 9.2 – Таблиця даних

x	0.10	0.15	0.20	0.25	0.30	0.35	0.40
y	0.995	0.988	0.980	0.969	0.955	0.939	0.921

Таблиця 9.3– Таблиця даних

x	0.65	0.70	0.75	0.80	0.85	0.90	0.95
y	0.605	0.644	0.681	0.71	0.75	0.783	0.813

Література

- 1 Чепак, Л.В. Численные методы. Использование MatLab [Текст] / Чепак, А.Г. Масловская. – Благовещенск, 2005 г. – 69 с.
- 2 Киреев, В.И. Численные методы в системах и задачах [Текст]: .пособие / В.И. Киреев, В.А. Пантелеев. – М.:Высш.шк., 2008 г. – 480 с.
- 3 Половко, А.М. Matlab для студента [Текст] / А.М. Половко, П.Н. сов. – Спб.: «БХВ-Петербург», 2005 г. – 317 с.

Розробив: Ланська С.С.

Розглянуто та схвалено

на засіданні циклової комісії

програмної інженерії

Протокол № 4 від 17.11.2021 р.

Голова комісії _____ Ланська С.С.

ЛАБОРАТОРНЕ ЗАНЯТТЯ №10

з навчальної дисципліни «Чисельні методи»

Тема	Обробка даних за допомогою методу найменших квадратів
Мета	Навчитися визначати параметри функціональних (емпіричних) залежностей методом найменших квадратів

Час –2 години

Теоретичні відомості

У процесі вивчення різних питань природознавства, економіки і техніки, соціології, педагогіки доводиться на основі великої кількості дослідних даних виявляти суттєві фактори, які впливають на досліджуваний об'єкт, а також встановлювати форму зв'язку між різними зв'язаними одна з одною величинами (ознаками).

Нехай у результаті досліджень дістали таку таблицю деякої функціональної залежності:

x_i	x_1	x_2	...	x_n
y_i	y_1	y_2	...	y_n

Треба знайти аналітичний вигляд функції y_i , яка добре відображала б цю таблицю дослідних даних. Функцію y_i можна шукати у вигляді інтерполяційного полінома. Але інтерполяційні поліноми не завжди добре відображають характер поведінки таблично заданої функції. До того ж значення y_i дістають у результаті експерименту, а вони, як правило, сумнівні. В такому разі задача інтерполювання табличної функції втрачає сенс. Тому шукають таку функцію $g(x)$, значення якої при $x=x_i$ досить близькі до табличних значень $y_1, \dots, y_n$. Формулу $y_i = g(x_i)$ називають емпіричною або рівнянням регресії y_i на x_i . Емпіричні формули мають велике практичне значення, вдало підібрана емпірична формула дає змогу не тільки апроксимувати сукупність експериментальних даних, «згладжуючи» значення величин y_i , а й екстраполювати знайдену залежність на інші проміжки значень x .

Процес побудови емпіричних формул складається з двох етапів: встановлення загального виду цієї формули і визначення найкращих її параметрів.

Щоб встановити вигляд емпіричної формули, на площині будують точки з координатами (x_i, y_i) , де $i = 1, 2, \dots, n$. Деякі з цих точок сполучають плавною кривою, яку проводять так, щоб вона проходила якомога ближче до всіх даних точок. Після цього візуально визначають, графік якої з відомих нам функцій найкраще підходить до побудованої кривої. Звичайно, намагаються підібрати найпростіші функції: лінійну, квадратичну, дробово-раціональну, степеневу, показникову, логарифмічну.

2.2 Метод найменших квадратів. Апроксимація поліномами

Нехай задана функція $f(x)$: $f(x_0) = y_0$; $f(x_1) = y_1$; ...; $f(x_n) = y_n$. Треба побудувати багаточлен $g_m(x)$ (обов'язково $m < n$) такий, щоб сума квадратів (тоді не треба враховувати знак) відхилень була мінімальна

$$S = \sum_{i=0}^n (g_m(x_i) - f(x_i))^2 \rightarrow \min, \quad (10.1)$$

тобто необхідно знайти екстремум (мінімум) суми квадратів (максимум нас нецікавить). Якщо функція $g_m(x)$ є алгебраїчний багаточлен

$$g_m(x) = a_0 + a_1x + a_2x^2 + \dots + a_mx^m,$$

то умова наближення (10.1) має вигляд:

$$S = \sum_{i=0}^n (a_0 + a_1x_i + a_2x_i^2 + \dots + a_mx_i^m - y_i)^2 \rightarrow \min \quad (10.2)$$

Необхідна умова мінімуму функції багатьох змінних (10.2)— її частинні похідні мають дорівнювати нулю, тобто

$$\begin{cases} \frac{\partial S}{\partial a_1} = 0, \frac{\partial S}{\partial a_2} = 0, \dots, \frac{\partial S}{\partial a_m} = 0; \\ \frac{\partial S}{\partial a_0} = 2 \sum_{i=0}^n (a_0 + a_1x_i + \dots + a_mx_i^m - y_i) \cdot 1 = 0 \\ \frac{\partial S}{\partial a_1} = 2 \sum_{i=0}^n (a_0 + a_1x_i + \dots + a_mx_i^m - y_i) \cdot x_i = 0 \\ \dots\dots\dots \\ \frac{\partial S}{\partial a_m} = 2 \sum_{i=0}^n (a_0 + a_1x_i + \dots + a_mx_i^m - y_i) \cdot x_i^m = 0 \end{cases} \quad (10.3)$$

Ця система лінійних алгебраїчних рівнянь визначає усі коефіцієнти a_i .

Система (10.3) називається системою нормальних рівнянь. Після рівносілних перетворень системи (10.3) маємо систему:

$$\begin{cases} a_0 \sum_{i=0}^n x_i^0 + a_1 \sum_{i=0}^n x_i^1 + \dots + a_m \sum_{i=0}^n x_i^m = \sum_{i=0}^n y_i \\ a_0 \sum_{i=0}^n x_i^1 + a_1 \sum_{i=0}^n x_i^2 + \dots + a_m \sum_{i=0}^n x_i^{m+1} = \sum_{i=0}^n y_i x_i \\ \dots\dots\dots \\ a_0 \sum_{i=0}^n x_i^m + a_1 \sum_{i=0}^n x_i^{m+1} + \dots + a_m \sum_{i=0}^n x_i^{m+m} = \sum_{i=0}^n y_i x_i^m \end{cases} \quad (10.4)$$

Систему нормальних рівнянь (10.4) можна записати за допомогою матриць:

$$\mathbf{S} \cdot \mathbf{A} = \mathbf{B}; \text{ де } \mathbf{A} = (a_0, a_1, \dots, a_m); \mathbf{S} = \{S_{kj}\}_{k,j=0}^m = \sum_{i=0}^n x_i^{k+j}; \mathbf{B} = \{b_k\}_{k=0}^m = \sum_{i=0}^n y_i x_i^k. \quad (10.5)$$

Апроксимація лінійним поліномом

Нехай між табличними даними $(x_i, y_i \quad i = 1, 2, \dots, n)$ існує лінійна залежність. Шукатимемо емпіричну формулу у вигляді $y = ax + b$, де коефіцієнти a і b невідомі.

Знайдемо значення a і b , за яких функція $S(a, b) = \sum_{i=1}^n (y_i - ax_i - b)^2$ матиме мінімальне значення. Щоб знайти ці значення, прирівняємо до нуля часткові похідні функції $S(a, b)$

$$\begin{cases} \frac{\partial S}{\partial a} = 2 \sum_{i=1}^n (y_i - ax_i - b)(-x_i) = 0, \\ \frac{\partial S}{\partial b} = 2 \sum_{i=1}^n (y_i - ax_i - b)(-1) = 0. \end{cases}$$

Звідси, врахувавши, що $\sum_{i=1}^n b = nb$, маємо

$$\begin{cases} a \sum_{i=1}^n x_i^2 + b \sum_{i=1}^n x_i = \sum_{i=1}^n x_i y_i, \\ a \sum_{i=1}^n x_i + nb = \sum_{i=1}^n y_i. \end{cases}$$

Розв'язавши відносно a і b останню систему, знайдемо

$$a = \frac{n \sum_{i=1}^n x_i y_i - \sum_{i=1}^n x_i \sum_{i=1}^n y_i}{n \sum_{i=1}^n x_i^2 - \left(\sum_{i=1}^n x_i \right)^2}, \quad b = \frac{\sum_{i=1}^n y_i \sum_{i=1}^n x_i^2 - \sum_{i=1}^n x_i y_i \sum_{i=1}^n x_i}{n \sum_{i=1}^n x_i^2 - \left(\sum_{i=1}^n x_i \right)^2}.$$

Блок схема методу найменших квадратів (МНК) представлено на рисунку 10.1.

Якщо ми маємо дві точки (x_0, y_0) та (x_1, y_1) , то рівняння прямої має вигляд (див. рис. 10.2).

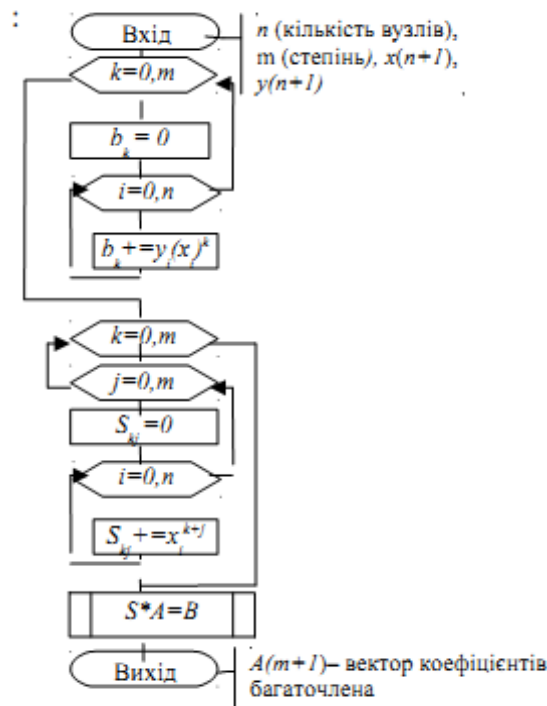


Рисунок 10.1– Блок-схема методу НК

$$y = y_0; \frac{x - x_0}{x_0 - x_1} + y \frac{x - x_0}{x_1 - x_0}.$$

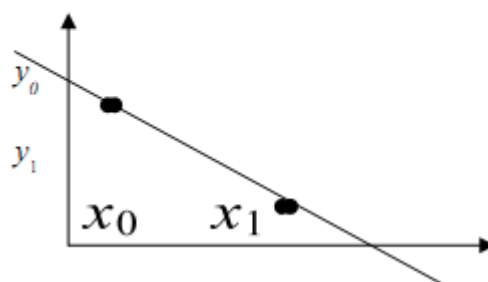


Рисунок 9.2 – Лінійна апроксимація

$p = \text{polyfit}(x, y, n)$

Функція $p = \text{polyfit}(x, y, n)$ знаходить коефіцієнти полінома $p(x)$ ступеня n , який апроксимує функцію $y(x)$ за допомогою методу найменших квадратів. Виходом є рядок p довжини $n+1$, яка містить коефіцієнти апроксимуючого полінома.

Хід роботи

- 1) Опрацювати теоретичний матеріал по літературі та конспекту.
- 2) Розробити алгоритм вирішення емпіричних залежностей, згідно з варіантом (табл.10.1), методом найменших квадратів. Визначити коефіцієнти лінійного рівняння регресії за експериментальними даними. Протестувати функції за допомогою функції *polyfit*.
- 3) Набрати, налагодити, отримати результати рішення та їх проаналізувати.
- 4) Захистити роботу.

Звіт повинен містити

- 1) Тему та мету лабораторної роботи.
- 2) Постановку задачі та рівняння, перевірити результати за допомогою функції *polyfit()*.
- 3) Основні формули метода та аналітичні розрахунки (скопіювати з Matlab).

Завдання для самостійної роботи

Згідно заданої таблиці (див. рис. 10.1) розробити алгоритм вирішення емпіричних залежностей за допомогою методу найменших квадратів.

Таблиця 10.1 – Варіанти завдань

Варіант									
1		2		3		4		5	
x	y	x	y	x	y	x	y	x	y
1.0	7.190	0.00	4.588	-5.0	17.663	2.0	0.190	0.000	2.994
1.2	7.466	0.25	4.197	-4.5	16.206	2.2	0.228	0.025	3.004
1.4	7.912	0.50	4.129	-4.0	16.360	2.4	0.246	0.050	3.012
1.6	8.495	0.75	3.066	-3.5	15.772	2.6	0.269	0.075	3.018
1.8	8.727	1.00	3.111	-3.0	14.087	2.8	0.300	0.100	3.017
2.0	8.842	1.25	2.305	-2.5	13.684	3.0	0.279	0.125	3.026
2.2	9.673	1.50	2.348	-2.0	13.080	3.2	0.334	0.150	3.033
2.4	9.622	1.75	1.414	-1.5	11.589	3.4	0.345	0.175	3.034
2.6	10.090	2.00	0.926	-1.0	11.050	3.6	0.365	0.200	3.037
2.8	10.744	2.25	0.659	-0.5	10.815	3.8	0.387	0.225	3.047
3.0	10.988	2.50	0.067	0.0	9.802	4.0	0.414	0.250	3.048
3.2	11.028	2.75	-0.818	0.5	8.603	4.2	0.441	0.275	3.055
3.4	11.897	3.00	-1.014	1.0	8.666	4.4	0.438	0.300	3.059
3.6	12.311	3.25	-1.389	1.5	7.927	4.6	0.456	0.325	3.065
3.8	12.737	3.50	-1.748	2.0	7.518	4.8	0.471	0.350	3.067
4.0	13.054	3.75	-2.831	2.5	5.723	5.0	0.513	0.375	3.076
4.2	13.166	4.00	-2.854	3.0	5.009	5.2	0.528	0.400	3.081
4.4	13.956	4.25	-3.215	3.5	5.112	5.4	0.562	0.425	3.080
4.6	13.949	4.50	-3.981	4.0	3.986	5.6	0.573	0.450	3.092
4.8	14.562	4.75	-4.299	4.5	2.822	5.8	0.584	0.475	3.096
5.0	15.312	5.00	-4.843	5.0	2.519	6.0	0.597	0.500	3.105

Продовження таблиці 10.1

6		7		8		9		10	
x	y	x	y	x	y	x	y	x	y
10	61.53	0.00	0.539	2.50	3.538	0.50	4.178	0.000	1.525
15	70.80	0.75	1.758	3.25	4.982	0.55	4.181	0.075	1.448
20	71.05	1.50	3.557	4.00	6.101	0.60	4.497	0.150	1.301
25	86.21	2.25	5.843	4.75	7.269	0.65	4.779	0.225	1.204
30	86.54	3.00	6.541	5.50	9.526	0.70	4.962	0.300	1.090
35	91.95	3.75	10.687	6.25	9.226	0.75	5.402	0.375	1.024
40	102.44	4.50	10.866	7.00	11.117	0.80	5.541	0.450	0.894
45	109.23	5.25	12.488	7.75	11.589	0.85	5.864	0.525	0.650
50	115.56	6.00	15.414	8.50	13.847	0.90	5.881	0.600	0.609
55	125.26	6.75	17.420	9.25	13.752	0.95	6.303	0.675	0.414
60	126.35	7.50	17.010	10.00	16.087	1.00	6.346	0.750	0.363
65	138.10	8.25	21.069	10.75	16.989	1.05	6.721	0.825	0.279
70	140.17	9.00	22.861	11.50	17.316	1.10	7.062	0.900	0.065
75	149.14	9.75	23.871	12.25	19.670	1.15	7.132	0.975	0.085
80	156.95	10.50	24.990	13.00	19.849	1.20	7.566	1.050	-0.055
85	161.53	11.25	28.548	13.75	21.478	1.25	7.702	1.125	-0.210
90	166.69	12.00	30.679	14.50	22.547	1.30	7.841	1.200	-0.299
95	172.72	12.75	31.546	15.25	23.634	1.35	8.295	1.275	-0.285
100	179.69	13.50	32.900	16.00	24.541	1.40	8.568	1.350	-0.561
105	184.44	14.25	34.477	16.75	25.110	1.45	8.818	1.425	-0.612
110	193.05	15.00	36.757	17.50	26.950	1.50	8.963	1.500	-0.781

Продовження таблиці 10.1

11		12		13		14		15	
x	y	x	y	x	y	x	y	x	y
-2.0	-3.47	0.0	3.63	0.0	9.199	0.0	1.758	0.00	3.83
-1.3	-3.42	0.1	4.09	0.4	7.628	0.8	1.522	0.15	3.45
-0.6	-3.01	0.2	4.54	0.8	8.223	1.6	1.375	0.30	7.46
0.1	-2.61	0.3	5.45	1.2	6.983	2.4	1.008	0.45	9.40
0.8	-2.12	0.4	5.82	1.6	6.021	3.2	0.537	0.60	11.90
1.5	-1.67	0.5	6.10	2.0	6.500	4.0	0.517	0.75	11.64
2.2	-1.38	0.6	7.50	2.4	5.309	4.8	0.056	0.90	14.18
2.9	-1.23	0.7	7.93	2.8	4.874	5.6	0.159	1.05	15.44
3.6	-1.00	0.8	8.01	3.2	3.907	6.4	-0.381	1.20	16.70
4.3	-0.47	0.9	9.24	3.6	4.091	7.2	-0.710	1.35	20.12
5.0	0.05	1.0	9.90	4.0	2.909	8.0	-0.946	1.50	21.49
5.7	-0.01	1.1	10.23	4.4	2.800	8.8	-1.174	1.65	22.15
6.4	0.56	1.2	10.21	4.8	2.063	9.6	-1.603	1.80	27.05
7.1	0.74	1.3	11.32	5.2	0.891	10.4	-1.837	1.95	25.64
7.8	1.53	1.4	11.85	5.6	0.746	11.2	-2.045	2.10	27.63
8.5	1.52	1.5	12.30	6.0	0.069	12.0	-2.211	2.25	32.91
9.2	1.87	1.6	13.18	6.4	-0.266	12.8	-2.524	2.40	32.52
9.9	2.25	1.7	13.51	6.8	-1.171	13.6	-2.599	2.55	35.05
10.6	2.50	1.8	14.56	7.2	-1.817	14.4	-3.189	2.70	35.35
11.3	2.96	1.9	14.96	7.6	-2.171	15.2	-3.280	2.85	39.94
12.0	3.22	2.0	15.60	8.0	-3.000	16.0	-3.421	3.00	40.91

Література

1 Чепак, Л.В. Численные методы. Использование MatLab [Текст] /

Л.В. Чепак, А.Г. Масловская. – Благовещенск, 2005 г. – 69 с.

2 Киреев, В.И. Численные методы в системах и задачах [Текст]:

учеб.пособие / В.И. Киреев, В.А. Пантелеев. – М.:Высш.шк., 2008 г. – 480 с.

3 Половко, А.М. Matlab для студента [Текст] / А.М. Половко, П.Н.

Бутусов. – Спб.: «БХВ-Петербург», 2005 г. – 317 с.

Розробив: Ланська С.С.

Розглянуто та схвалено

на засіданні циклової комісії

програмної інженерії

Протокол № 4 від 17.11.2021 р.

Голова комісії _____ Ланська С.С.

ЛАБОРАТОРНЕ ЗАНЯТТЯ №11

з навчальної дисципліни «Чисельні методи»

Тема	Обчислення інтегралів за допомогою формул прямокутників та трапецій
Мета	Обчислення інтегралів за допомогою формул прямокутників та трапецій в системі MATLAB

Час –2 години

Теоретичні відомості

Найбільш простим методом наближеного обчислення інтеграла є метод прямокутників, заснований на безпосередньому визначенні інтеграла:

$$\int_a^b f(x)dx = \lim_{n \rightarrow \infty} \sum_{i=1}^{n-1} f(\xi_i) \Delta x_i, \text{ де } \lim_{n \rightarrow \infty} \sum_{i=1}^{n-1} f(\xi_i) \Delta x_i - \text{є інтегральна сума, відповідна}$$

деякому розбиттю відрізаня $[a, b]$ і деякому вибору ξ_i на відрізках розбиття.

Обчислення певного інтеграла $I = \int_a^b f(x)dx$ геометрично зводиться до

обчислення площі криволінійної трапеції, обмеженою функцією $f(x)$, віссю абсцис і прямими $x=a$ і $x=b$ (див. рис. 11.1 а). Обчислимо наближене значення інтеграла таким чином. Замінімо криволінійну трапецію $DEba$ прямокутником $Avba$, провівши AB так, щоб фігури DAC і CEB вийшли приблизно рівній площі. Тоді площі криволінійної трапеції, обмеженій графіком функції $y=f(x)$, і отриманого прямокутника $Avba$ будуть приблизно рівні.

Враховуючи, що висота прямокутника $Avba$ є значення функції в крапці, запишемо наступну наближену рівність:

$$\int_a^b f(x)dx \approx (b-a)f(\xi)$$

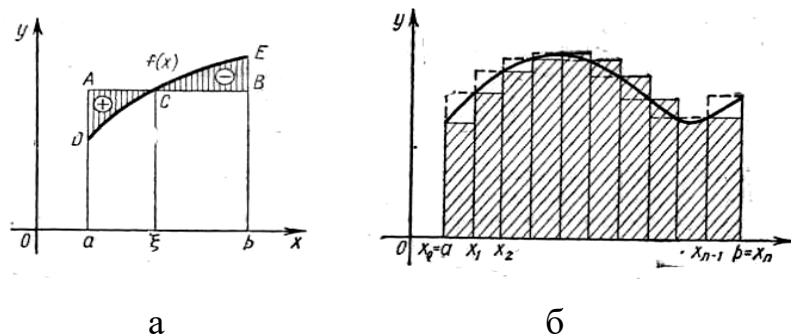


Рисунок 11.1 – Геометрична інтерпретація методу прямокутників

Для збільшення точності чисельної інтеграції можна відрізок $[a, b]$ розбити на декілька частин і для кожної з них обчислити наближене значення площі криволінійної трапеції, підставою якої є відрізок $x_i = x_{i+1} - x_i$ ($i=0, 1, n-1$), а заввишки — число, тобто значення функції в крапці $\xi_i \in [x_i, x_{i+1}]$ вибране з умови мінімуму помилки інтеграції. Тоді за наближене значення інтеграла на відрізку $[a, b]$ приймають інтегральну суму:

$$\int_a^b f(x)dx \approx f(\xi_0)\Delta x_0 + f(\xi_1)\Delta x_1 + f(\xi_2)\Delta x_2 + \dots + f(\xi_{n-1})\Delta x_{n-1}$$

Практично зручно ділити відрізок $[a, b]$ на рівні частини, а точки ($i = 0, 1, 2, \dots, n-1$) поєднувати із лівими $[I_L]$ або правими $[I_n]$ кінцями відрізків розбиття.

Якщо крапку ξ_i сумістити з лівим кінцем відрізка x_i , то наближене значення інтеграла геометрично дорівнює площі заштрихованої ступінчастої фігури (див. рис. 11.1 б), і може бути представлено формулою лівих прямокутників:

$$I_L = \int_a^b f(x)dx \approx \frac{b-a}{n}(y_0 + y_1 + \dots + y_{n-1}) = h \sum_{i=0}^{n-1} y_i \quad (11.1)$$

де $h=(b-a)/n$ — крок.

Якщо ж як крапка ξ_i вибрати правий кінець відрізка x_i , то наближене значення інтеграла графічно дорівнює площі ступінчастої фігури, обмеженою зверху пунктирною лінією, і обчислюється за формулою правих прямокутників:

$$I_{II} = \int_a^b f(x)dx \approx \frac{b-a}{n}(y_1 + y_2 + \dots + y_n) = h \sum_{i=1}^n y_i \quad (11.2)$$

Метод трапецій. Наближене значення певного інтеграла можна обчислити і іншим способом. Замінімо на відрізку $[a, b]$ дугу AB графіка підінтегральної функції $y=f(x)$ хордою (див. рис. 11.2, а), що стягує її, і обчислимо площу трапеції $Avba$. Набудемо значення певного інтеграла чисельно рівним площі цієї трапеції:

$$\int_a^b f(x)dx \approx (b-a) \frac{f(a) + f(b)}{2} \quad (11.3)$$

Це і є формула трапецій для наближеного обчислення інтеграла.

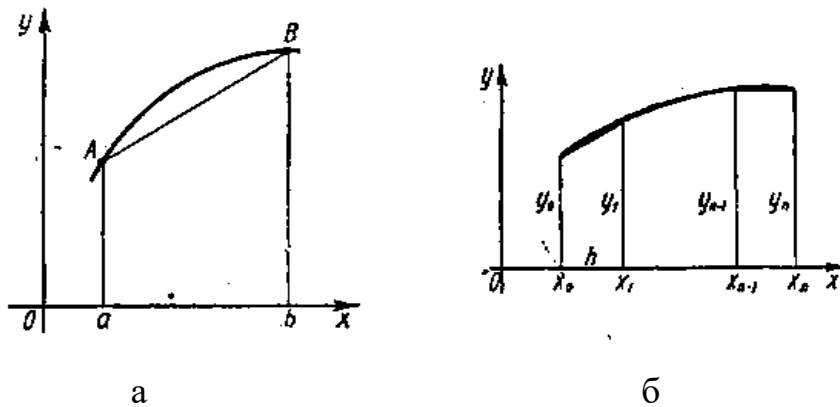


Рисунок 11.2 – Геометричний зміст методу трапецій

Точність обчислень зростає, якщо відрізок $[a, b]$ розділити на декілька частин і застосувати формулу трапецій до кожного відрізка Sx_i (рис. 11.2, б). Тоді

$$\int_{x_i}^{x_{i+1}} f(x)dx \approx \frac{f(x_i) + f(x_{i+1})}{2} \Delta x_i$$

Для простоти обчислень зручно ділити відрізок $[a, b]$ на рівні частини, в цьому випадку довжина кожного з відрізків розбиття є $x_i=(b-a)/n$. Чисельне значення інтеграла на відрізку x_i рівно:

$$\int_{x_i}^{x_{i+1}} f(x)dx \approx \frac{b-a}{n} \cdot \frac{f(x_i) + f(x_{i+1})}{2}$$

а на всьому відрізку $[a, b]$ відповідно:

$$\int_a^b f(x)dx \approx \frac{b-a}{n} \sum_{i=0}^{n-1} \frac{f(x_i) + f(x_{i+1})}{2} = \frac{b-a}{n} \sum_{i=0}^{n-1} \frac{y_i + y_{i+1}}{2}$$

Оскільки під знаком суми величини y_i зустрічаються двічі (від $i=1$ до $i=n-1$), то остання рівність запишеться таким чином:

$$\int_a^b f(x)dx \approx \frac{b-a}{n} \left(\frac{y_0}{2} + y_1 + y_2 + \dots + y_{n-1} + \frac{y_n}{2} \right) = \frac{b-a}{n} \left[\sum_{i=1}^{n-1} y_i + \frac{y_0 + y_n}{2} \right] \quad (11.4)$$

Ця формула називається загальною формулою трапецій. Загальну формулу трапецій можна переписати у вигляді:

$$\int_a^b f(x)dx \approx \frac{h}{2} (y_0 + 2y_1 + 2y_2 + \dots + 2y_{n-1} + y_n) = \frac{b-a}{n} \left[\sum_{i=1}^{n-1} y_i + \frac{y_0 + y_n}{2} \right] \quad (11.5)$$

де крок

$$h = \frac{b-a}{2} \quad (11.6)$$

Функції Matlab:

$I = \text{trapz}(x, y)$

$I = \text{trapz}(y)$

Функція $I = \text{trapz}(x, y)$ обчислює інтеграл від функції y по змінній x , використовуючи метод трапецій. Аргументи x і y можуть бути одновимірними масивами однакового розміру, або масив Y може бути двовимірним, але тоді має виконуватися умова $\text{size}(Y, 1) = \text{length}(x)$. В останньому випадку обчислюється інтеграл для кожного стовпчика.

Функція $I = \text{trapz}(y)$ обчислює інтеграл, припускаючи, що крок інтегрування постійний і дорівнює одиниці; в разі, коли крок h відмінний від одиниці, але постійний, досить обчислений інтеграл помножити на h .

Приклад:

Обчислимо інтеграл: $I = \int_0^x \sin(x) dx$.

Його точне значення дорівнює двом.

Виберемо рівномірну сітку

$x = 0:\pi/100:\pi;$

$y = \sin(x);$

тоді обидва інтеграла

$I = \text{trapz}(x, y)$ и

$I = \pi/100 * \text{trapz}(y)$

дають однаковий результат:

$I = 1.9998.$

Хід роботи

- 1) Опрацювати теоретичний матеріал по літературі та конспекту.
- 2) Розробити алгоритм обчислення інтегралу, згідно завдання (табл. 11.1), за допомогою формул прямокутників та трапецій.
- 3) Набрати, налагодити та отримати результати рішення, їх проаналізувати.
- 4) Перевірити результати виконання роботи за допомогою функції `trapz()`.
- 5) Захистити роботу.

Звіт повинен містити

- 1) Тему та мету лабораторної роботи.
- 2) Постановку задачі та рівняння.
- 3) Основні формули методу та аналітичні розрахунки (скопювати з MATLAB).
- 4) Перевірити результати за допомогою функції.

Завдання для самостійної роботи

Скласти програму обчислення інтегралу від заданої функції на відрізку $[a, b]$ по формулам прямокутників та трапецій з шагом $h=0,1; 0,05$. Порівняти результати. Початкові дані для виконання завдання беруться з таблиці.

Таблиця 11.1 – Варіанти завдань

№ п/п	Функція	a	b
1	$0.37e^{\cos x}$	0	1
2	$0.5x + x \ln x$	1	2
3	$(x + 1.9)\sin(x/3)$	1	2
4	$\frac{1}{x} \ln(x + 2)$	2	3
5	$\frac{3\cos x}{2x + 1.7}$	0	1
6	$(2x + 0.6)\cos(x/2)$	1	2
7	$2.6x^2 \ln x$	1.2	2.2
8	$(x^2 + 1)\sin(x - 0.5)$	1	2

Продовження таблиці 11.1

№ п/п	Функція	a	b
9	$x^2 \cos(x/4)$	2	3
10	$\frac{\sin(0.2x - 3)}{x^2 + 1}$	3	4
11	$\frac{2 \sin(x + 0.2)}{2x}$	0	1
12	$2\cos(x) - x^2$	2	3
13	$x^2 \sin(x/4)$	0	1
14	$\frac{\cos(x)}{1 + x^2}$	1	2
15	$1 + \frac{\sin(x)}{x^2}$	2	3

Література

1 Чепак, Л.В. Численные методы. Использование MatLab [Текст] /

Л.В. Чепак, А.Г. Масловская. – Благовещенск, 2005 г. – 69 с.

2 Киреев, В.И. Численные методы в системах и задачах [Текст]:

учеб.пособие / В.И. Киреев, В.А. Пантелеев. – М.:Высш.шк., 2008 г. – 480 с.

3 Половко, А.М. Matlab для студента [Текст] / А.М. Половко, П.Н.

Бутусов. – Спб.: «БХВ-Петербург», 2005 г. – 317 с.

Розробив: Ланська С.С.

Розглянуто та схвалено

на засіданні циклової комісії

програмної інженерії

Протокол № 4 від 17.11.2021 р.

Голова комісії _____ Ланська С.С.

ЛАБОРАТОРНЕ ЗАНЯТТЯ №12

з навчальної дисципліни «Чисельні методи»

Тема	Обчислення інтегралів за допомогою формули Сімпсона
Мета	Вміти обчислювати інтеграли за допомогою формули Сімпсона в системі MATLAB

Час – 2 години

Теоретичні відомості

Точність наближеної інтеграції помітно зросте, якщо підінтегральну функцію $y=f(x)$ на відрізку $[a, b]$ замінити квадратичною функцією (див. рис. 12.1), що приймає у вузлах $x_0 = a, x_1, x_2 = b$ значення $y_0=f(x_0), y_1=f(x_1)$ і $y_2 = f(x_2)$

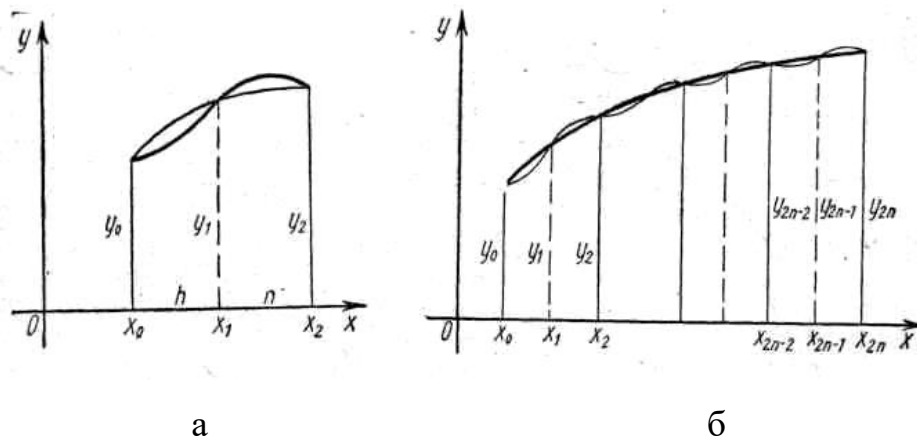


Рисунок 12.1 – Геометричний зміст методу Сімпсона

Як інтерполяційний многочлен скористаємося многочленом Ньютона

$$P_2(x) = y_0 + \frac{\Delta y_0}{h}(x - x_0) + \frac{\Delta^2 y_0}{2h^2}(x - x_0)(x - x_1)$$

Тоді

$$\begin{aligned} \int_{x_0}^{x_2} f(x) dx &= \int_{x_0}^{x_2} P_2(x) dx = \left(2h y_0 + \frac{\Delta y_0}{h} \cdot \frac{4h^2}{2} + \frac{\Delta^2 y_0}{2h^2} \cdot \frac{8h^3}{3} - \frac{\Delta^2 y_0}{2h^2} \cdot h \frac{4h^2}{2} \right) = \\ &= 2h y_0 + 2h \Delta y_0 + \frac{h}{3} \Delta^2 y_0 = \frac{h}{3} (y_0 + 4y_1 + y_2) \end{aligned} \quad (12.1)$$

Співвідношення (12.1) називається формулою Сімпсона. Формула Сімпсона володіє підвищеною точністю і є точною не тільки для многочленів другого ступеня, але і третьою.

Погрішність формули Сімпсона

$$R = \int_{x_0}^{x_0+2h} f(x)dx - \frac{h}{3}(y_0 + 4y_1 + y_2)$$

оцінюється таким чином:

$$R = -\frac{h^5}{90} y^{iv}(\xi) \quad (12.2)$$

де крапка $\xi \in [x_0, x_0+2h]$.

Для збільшення точності обчислень відрізок $[a, b]$ розбивають на n пару ділянок $[x_{2n-2}, x_{2n-1}, x_{2n}]$ (рис. 12.1, б) і замінюючи підінтегральну функцію інтерполяційним многочленом Ньютона другого ступеня, набувають наближеного значення інтеграла на кожній ділянці довжини $2h$:

$$\int_{x_0}^{x_2} f(x)dx \approx \frac{h}{3}(y_0 + 4y_1 + y_2)$$

$$\int_{x_2}^{x_4} f(x)dx \approx \frac{h}{3}(y_2 + 4y_3 + y_4)$$

.....

$$\int_{x_{2n-2}}^{x_{2n}} f(x)dx \approx \frac{h}{3}(y_{2n-2} + 4y_{2n-1} + y_{2n})$$

Тоді чисельне значення певного інтеграла на відрізку $[a, b]$ дорівнюватиме сумі інтегралів, тобто

$$\int_a^b f(x)dx \approx \frac{h}{3}[(y_0 + y_{2n}) + 4(y_1 + \dots + y_{2n-1}) + (y_2 + \dots + y_{2n-2})] \quad (12.3)$$

Співвідношення (12.3) називається загальною формулою Сімпсона, Її можна записати також у вигляді

$$\int_a^b f(x)dx \approx \frac{b-a}{6n}[(y_0 + y_{2n}) + 4(y_1 + \dots + y_{2n-1}) + (y_2 + \dots + y_{2n-2})] \quad (12.4)$$

де $h = \frac{b-a}{2n}$.

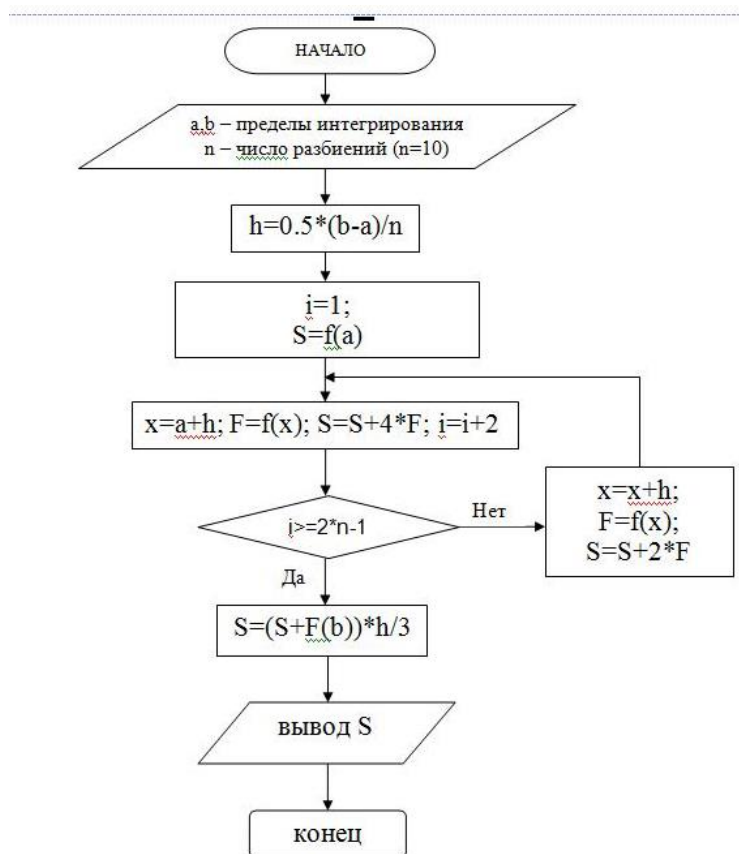


Рисунок 12.2 – Блок-схема вирішення алгоритму

Обчислення інтегралу методом квадратур здійснюється процедурою

$$[I, cnt] = quad(' <имя функции> ', a, b).$$

Тут a і b – нижня і верхня межа зміни аргументу функції; I – отримане значення інтеграла; cnt – кількість звернень до обчислення функції, представленої М-файлом з назвою, зазначеним в $<ім'я функції>$.

Функція `quad` використовує квадратурні формули Ньютона-Котеса четвертого порядку.

Хід роботи

- 1) Опрацювати теоретичний матеріал по літературі та конспекту.
- 2) Розробити алгоритм обчислення інтегралу, згідно завдання (табл. 12.1), за допомогою формули Сімпсона.
- 3) Набрати, налагодити, отримати результати рішення та проаналізувати їх за допомогою функції `quad()`.
- 4) Захистити роботу.

Звіт повинен містити

- 1) Тему та мету лабораторної роботи.
- 2) Постановку задачі та рівняння.
- 3) Основні формули методів та аналітичні розрахунки (скопіювати з MATLAB).

Завдання для самостійної роботи:

Скласти програму обчислення інтегралу від заданої функції на відрізку $[a, b]$ по формулі Сімпсона з шагом $h=0,1; 0,05$. Порівняти результати. Початкові дані для виконання завдання беруться з таблиці.

Таблиця 12.1 – Варіанти завдань

№ п/п	Функція	a	b
1	$0.37e^{\cos x}$	0	1
2	$0.5x + x \ln x$	1	2
3	$(x + 1.9) \sin(x/3)$	1	2
4	$\frac{1}{x} \ln(x + 2)$	2	3
5	$\frac{3 \cos x}{2x + 1.7}$	0	1
6	$(2x + 0.6) \cos(x/2)$	1	2
7	$2.6x^2 \ln x$	1.2	2.2
8	$(x^2 + 1) \sin(x - 0.5)$	1	2
9	$x^2 \cos(x/4)$	2	3
10	$\frac{\sin(0.2x - 3)}{x^2 + 1}$	3	4
11	$\frac{2 \sin(x + 0.2)}{2x}$	0	1
12	$2 \cos(x) - x^2$	2	3
13	$x^2 \sin(x/4)$	0	1
14	$\frac{\cos(x)}{1 + x^2}$	1	2
15	$1 + \frac{\sin(x)}{x^2}$	2	3

Література

- 1 Чепак, Л.В. Численные методы. Использование MatLab [Текст] /
Л.В. Чепак, А.Г. Масловская. – Благовещенск, 2005 г. – 69 с.
- 2 Киреев, В.И. Численные методы в системах и задачах [Текст]:
учеб.пособие / В.И. Киреев, В.А. Пантелеев. – М.:Высш.шк., 2008 г. – 480 с.
- 3 Половко, А.М. Matlab для студента [Текст] / А.М. Половко, П.Н.
Бутусов. – Спб.: «БХВ-Петербург», 2005 г. – 317 с.

Розробив: Ланська С.С.

Розглянуто та схвалено

на засіданні циклової комісії

програмної інженерії

Протокол № 4 від 17.11.2021 р.

Голова комісії _____ Ланська С.С.

ЛАБОРАТОРНЕ ЗАНЯТТЯ №13

з навчальної дисципліни «Чисельні методи»

Тема	Інтегрування диференціальних рівнянь методом Ейлера
Мета	Вміти інтегрувати диференціальні рівняння методом Ейлера в системі MATLAB

Час –2 години

Теоретичні відомості

Нехай дано диференціальне рівняння першого порядку

$$y' = f(x, y) \quad (13.1)$$

початковою умовою

$$x = x_0, y(x_0) = y_0 \quad (13.2)$$

Потрібно знайти вирішення рівняння (13.1) на відрізку $[a, b]$.

Розіб'ємо відрізок $[a, b]$ на n рівних частин і отримаємо послідовність $x_0, x_1, \dots, x_n$, де $x_i = x_0 + ih$ ($i = 0, 1, 2, \dots, n$), а $h = (b-a)/n$ — крок інтеграції.

Виберемо k -й ділянку $[x_k, x_{k+1}]$ і проінтегруємо рівняння (13.1):

$$\int_{x_k}^{x_{k+1}} f(x, y) dx = \int_{x_k}^{x_{k+1}} y' dx = y(x) \Big|_{x_k}^{x_{k+1}} = y(x_{k+1}) - y(x_k) = y_{k+1} - y_k$$

тобто

$$y_{k+1} = y_k + \int_{x_k}^{x_{k+1}} f(x, y) dx \quad (13.3)$$

Якщо в останньому інтегралі підінтегральну функцію на ділянці $[x_k, x_{k+1}]$ прийняти постійною і рівною початковому значенню в точці $x = x_k$, то отримаємо

$$\int_{x_k}^{x_{k+1}} f(x_k, y_k) dx = f(x_k, y_k) \cdot x \Big|_{x_k}^{x_{k+1}} = f(x_k, y_k)(x_{k+1} - x_k) = y'_k h$$

Тоді формула (13.3) набере вигляду

$$y_{k+1} = y_k + y'_k h \quad (13.3')$$

Позначивши $y_{k+1} - y_k = \Delta y_k$, тобто $y'_k h = \Delta y_k$, отримаємо:

$$y_{k+1} = y_k + \Delta y_k \quad (13.4)$$

Продовжуючи цей процес і кожного разу приймаючи підінтегральну функцію на відповідній ділянці постійною і рівною її значенню на початку ділянки, отримаємо таблицю вирішень диференціального рівняння на заданому відрізку $[a, b]$. Рівність (13.4) означає, що на відрізку $[x_k, x_{k+1}]$ інтегральна крива $y = y(x)$ приблизно замінюється прямолінійним відрізком, що виходить з точки $M_k(x_k, y_k)$ з кутовим коефіцієнтом $f(x_k, y_k)$. Як наближення шуканої інтегральної кривої отримуємо ламану лінію з вершинами в точках $M_0(x_0; y_0), M_1(x_1; y_1), \dots, M_n(x_n; y_n)$. Перша ланка стосується дійсної інтегральної кривої в точці $M_0(x_0; y_0)$ (рисунок 13.1).

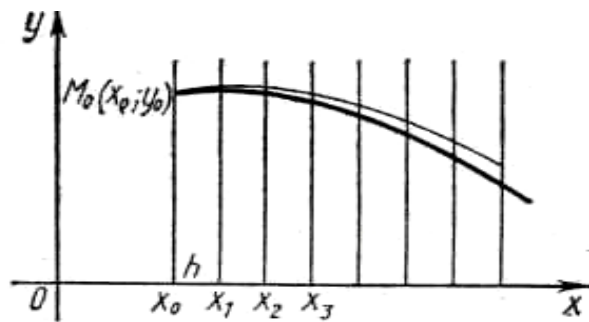


Рисунок 13.1 – Інтегральна крива

Якщо функція $f(x, y)$ в деякому прямокутнику

$$R\{|x - x_0| \leq a, |y - y_0| \leq b\}$$

задовольняє умові

$$|f(x_1, y_1) - f(x_1, y_2)| \leq N|y_1 - y_2| \quad (N = \text{const}) \quad (13.5)$$

і, крім того

$$\left| \frac{df}{dx} \right| = \left| \frac{\partial f}{\partial x} + f \frac{\partial f}{\partial y} \right| \leq M, (M = \text{const}) \quad (13.6)$$

то має місце наступна оцінка погрішності:

$$|y(x_n) - y_n| \leq \frac{hM}{2N} [(1 + hN)^n - 1] \quad (13.7)$$

де $y(x_n)$ — значення точного вирішення рівняння (13.1) при $x = x_n$, а y_n — наближене значення, отримане на n -му кроці.

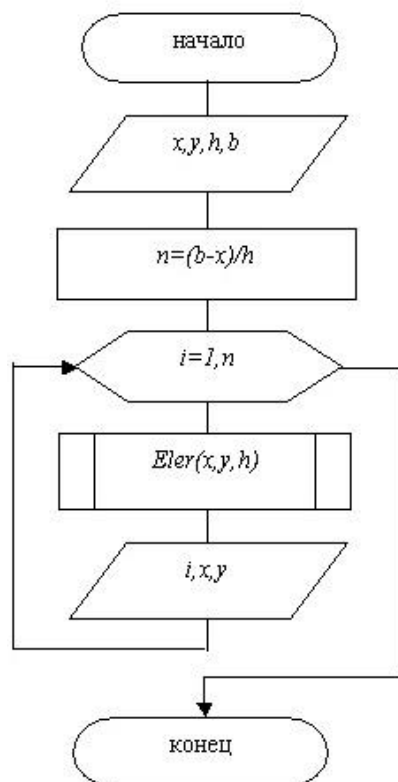
Формула (13.7) має в основному теоретичне застосування. На практиці, як правило, застосовують «подвійний прорахунок». Спочатку розрахунок

ведеться з кроком h , потім крок дроблять і повторний розрахунок ведеться з кроком $h/2$. Погрішність точнішого значення y^*n оцінюється формулою:

$$|y^*n - y(x_n)| \approx |y^*n - y_n|$$

Алгоритм методу Ейлера можна побудувати у вигляді двох програмних модулів: основної програми і підпрограми ELER, що реалізує метод (див. рис. 13.2).

Головна програма



Підпрограма Eler

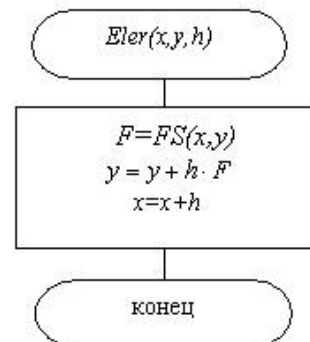


Рисунок 13.2 – Схема алгоритму методу Ейлера

Тут:

(x, y) – при введенні початкова точка, далі поточні значення табличній функції;

h – крок інтегрування диференціального рівняння;

b – кінець інтервалу інтегрування.

Бібліотека MATLAB включає кілька функцій, що реалізують різні методи розв'язання задачі Коші для звичайних диференціальних рівнянь (ode, ordinary differential equations). Синтаксично ці функції розрізняються лише

іменами (точніше кажучи, алфавітно-цифровими добавками до символів *ode*), спосіб звернення до них однаковий. ці функції використовують методи різного порядку (іноді метод і / або порядок варіюється від кроку до кроку), є й інші внутрішні відмінності.

У функції *ode45* використовується явний метод Рунге-Кутта 4-го і 5-го порядків в модифікації Дорманда і Принца. Якщо характеристики завдання невідомі, рекомендується першу спробу рішення зробити за допомогою цієї функції.

У функції *ode23* використовується явний метод Рунге-Кутта 2-го і 3-го порядків в модифікації Богацького і Шампіна. Якщо не потрібна велика точність і завдання її не дуже жорстке, ця функція може виявитися більш ефективною, ніж *ode45*.

У наведеному нижче описі ім'я функції дано узагальнено у вигляді *ode ****, де *** – будь-який з наведених вище алфавітно-цифрових суфіксів.

Найпростіше звернення до будь-якої функції *ode **** має наступний вигляд: $[tout, yout] = ode\ *** (fun, tspan, y0)$.

Тут:

- *fun* – покажчик на функцію обчислення правих частин диференційного рівняння;

- *tspan* – вектор, що містить "контрольні значення" незалежної змінної; мінімальний варіант $tspan = [t0\ tfinal]$ (початкове і кінцеве значення незалежної змінної), але можуть бути задані і проміжні значення, тоді $tspan = [t0\ t1\ \dots\ tfinal]$;

- *y0* – початкове значення залежної змінної (скаляр або вектор-стовпець);

- *tout* – вектор-стовпець контрольних значень незалежної змінної, якщо використовується мінімальний варіант для *tspan*, видаються все значення, які виходили в процесі чисельного інтегрування; якщо *tspan* містить і інші значення крім *t0* і *tfinal*, то $tout = tspan$;

– *yout* – рішення, представлене масивом, в якому кожен рядок відповідає одному елементу в стовпці *tout*.

Хід роботи

- 1) Опрацювати теоретичний матеріал по літературі та конспекту.
- 2) Розробити алгоритм вирішення диференціального рівняння методом Ейлера, згідно варіанта (табл. 13.1).
- 3) Набрати, налагодити, отримати результати рішення, та проаналізувати їх за допомогою солверу *ode23* або *ode45*.
- 4) Захистити роботу.

Звіт повинен містити

- 1) Тему та мету лабораторної роботи.
- 2) Постановку задачі та рівняння.
- 3) Основні формули метода та розрахунки за ними (скопіювати з MATLAB).
- 4) Результати розрахунків зроблених вручну.

Завдання для самостійної роботи:

Написати програму рішення диференціального рівняння $y' = f(x, y)$ методом Ейлера на відріжку $[a, b]$ з кроком h та $2h$ і початковою умовою $y(a) = y_0$. Початкові дані для виконання завдання беруться з таблиці 13.1. Побудувати графік.

Таблиця 13.1 – Варіанти завдань

№ п/п	Функція	a	b	y_0	h
1	$\frac{3x-y}{x^2+y}$	2	3	1	0.1
2	$\frac{2x+y+4}{2y+x}$	3	4	1	0.1
3	$\frac{x^2-y}{2x+y+1}$	0	1	2	0.1
4	$\frac{x^2-y+2}{xy+3x}$	2	3	1	0.1
5	$\frac{3-x-y^2}{2-xy^2}$	1	2	1	0.1
6	$\frac{2-x-y^2x}{3x+y}$	0	1	1	0.1
7	$\frac{1+3xy}{5-x+y^2}$	0	1	2	0.1
8	$\frac{x^2y+2}{2x-y}$	0	1	1	0.1
9	$\frac{x^2+y+2}{2x-y}$	2	3	2	0.1
10	$\frac{xy+4}{2y-xy+1}$	0	1	3	0.1
11	$y-4*x+3$	0	1	3	0.1
12	$\frac{x-y}{x^2-2}$	0	1	1	0.1
13	$4x-2+y$	1	2	2	0.1
14	$\frac{x-4y}{x^2}$	1	2	1	0.1
15	$xy-2\frac{x}{y^2}$	1	2	1	0.1

Література

1 Чепак, Л.В. Численные методы. Использование MatLab [Текст] /

Л.В. Чепак, А.Г. Масловская. – Благовещенск, 2005 г. – 69 с.

2 Киреев, В.И. Численные методы в системах и задачах [Текст]:

учеб.пособие / В.И. Киреев, В.А. Пантелеев. – М.:Выш.шк., 2008 г. – 480 с.

3 Половко, А.М. Matlab для студента [Текст] / А.М. Половко, П.Н.

Бутусов. – Спб.: «БХВ-Петербург», 2005 г. – 317 с.

Розробив: Ланська С.С.

Розглянуто та схвалено

на засіданні циклової комісії

програмної інженерії

Протокол № 4 від 17.11.2021 р.

Голова комісії _____ Ланська С.С.

ЛАБОРАТОРНЕ ЗАНЯТТЯ №14

з навчальної дисципліни «Чисельні методи»

Тема Розв'язання диференціальних рівнянь методом Рунге-Куты

Мета Вміти інтегрувати диференціальні рівняння методом Рунге-Куты
в системі MATLAB

Час –2 години

Теоретичні відомості

на відрізку $[a, b]$ потрібно знайти чисельне вирішення рівняння

$$y' = f(x, y) \quad (14.1)$$

початковою умовою

$$y(x_0) = y_0 \quad (14.2)$$

Розіб'ємо відрізок $[a, b]$ на n рівних частин точками $x_i = x_0 + ih$ ($i = 0, 1, 2, \dots, n$), де $h = (b-a)/n$ - крок інтеграції. У методі Рунге – Куты послідовні значення y_i шуканої функції у визначаються по формулі

$$y_{i+1} = y_i + \Delta y_i \quad (14.3)$$

Якщо розкласти функцію в ряд Тейлора і обмежитися членами до h^4 включно, то приріст функції y можна представити у вигляді

$$y = y(x+h) - y(x) = hy'(x) + \frac{h^2}{2} y''(x) + \frac{h^3}{6} y'''(x) + \frac{h^4}{24} y^{IV}(x) \quad (14.4)$$

де похідні $y''(x)$, $y'''(x)$, $y^{IV}(x)$ визначаються послідовним диференціюванням з рівняння (14.4).

Замість безпосередніх за формулою (14.4) в методі Рунге – Куты визначаються чотири числа:

$$\begin{aligned} k_1 &= hf(x; y) \\ k_2 &= hf\left(x + \frac{h}{2}; y + \frac{k_1}{2}\right) \\ k_3 &= hf\left(x + \frac{h}{2}; y + \frac{k_2}{2}\right) \\ k_4 &= hf(x+h; y+k_3) \end{aligned} \quad (14.5)$$

Можна довести, що якщо числам k_1, k_2, k_3, k_4 надати відповідно ваги $1/6; 1/3; 1/3; 1/6$, то середньозважене цих чисел, т. е. $\frac{1}{6} k_1 + \frac{1}{3} k_2 + \frac{1}{3} k_3 + \frac{1}{6} k_4$ з точністю до четвертих ступенів дорівнює значенню Δy , обчисленому за формулою (14.5):

$$\Delta y = \frac{1}{6} k_1 + \frac{1}{3} k_2 + \frac{1}{3} k_3 + \frac{1}{6} k_4 \quad (14.6)$$

Т.ч., для кожної пари поточних значень x_i і y_i по формулах (14.5) визначаються значення

$$\begin{aligned} k_1^{(i)} &= hf(x_i; y_i) \\ k_2^{(i)} &= hf\left(x_i + \frac{h}{2}; y_i + \frac{k_1^{(i)}}{2}\right) \\ k_3^{(i)} &= hf\left(x_i + \frac{h}{2}; y_i + \frac{k_2^{(i)}}{2}\right) \\ k_4^{(i)} &= hf(x_i + h; y_i + k_3^{(i)}) \end{aligned} \quad (14.7)$$

по формулі (14.6) знаходиться

$$\Delta y = \frac{1}{6} (k_1^{(i)} + 2k_2^{(i)} + 2k_3^{(i)} + k_4^{(i)})$$

і потім

$$y_{i+1} = y_i + \Delta y_i$$

Алгоритм методу Рунге-Кутта (4-го порядку) можна побудувати у вигляді двох програмних модулів: основної програми і підпрограми Rk4, що реалізує метод (див. рис. 14.1).

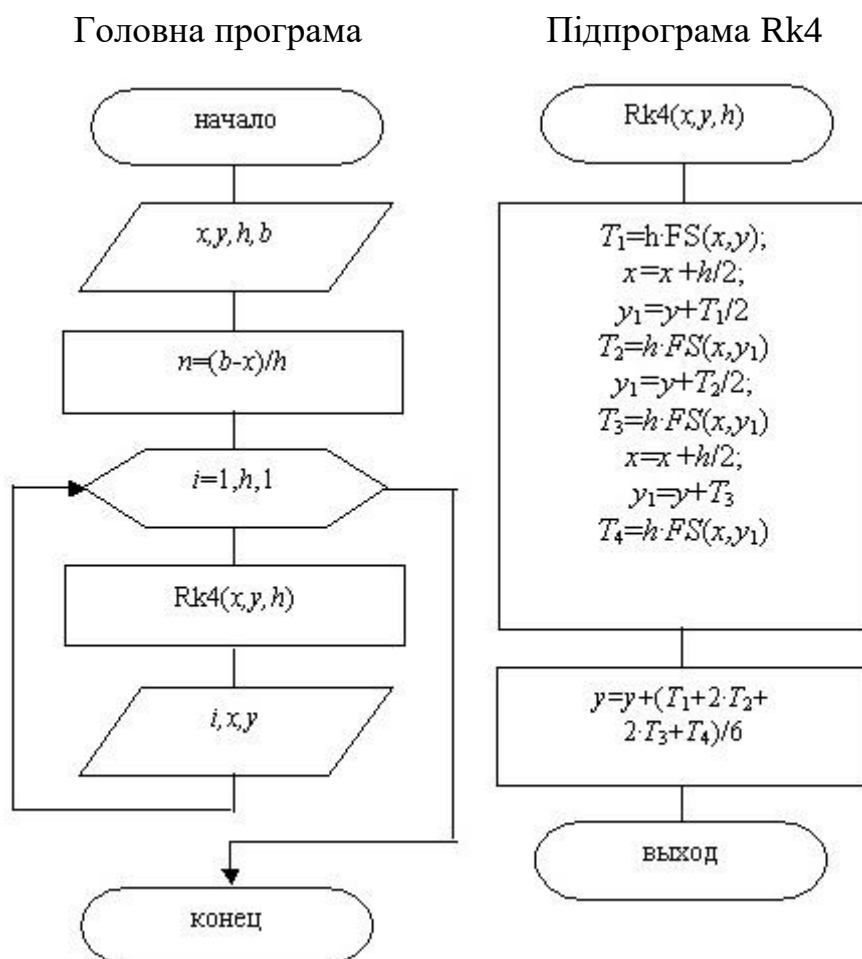


Рисунок 14.1 – Схема алгоритму методу Рунге-Кутта 4-го порядку.

Тут

(x, y) – при введенні початкова точка, далі поточні значення табличній функції;

h – крок інтегрування диференціального рівняння;

b – кінець інтервалу інтегрування.

Хід роботи

- 1) Опрацювати теоретичний матеріал по літературі та конспекту.
- 2) Розробити алгоритм вирішення диференціального рівняння методом Рунге-Кута, згідно з варіантом (табл. 14.1).
- 3) Набрати, налагодити, отримати результати рішення та проаналізувати їх за допомогою функцій *ode23* або *ode45*.
- 4) Захистити роботу.

Звіт повинен містити

- 1) Тему та мету лабораторної роботи.
- 2) Постановку задачі та рівняння.
- 3) Основні формули метода та аналітичні розрахунки (скопіювати з MATLAB).
- 4) Результати розрахунків зроблених вручну.

Завдання для самостійної роботи:

Написати програму рішення диференціального рівняння $y' = f(x, y)$ методом Рунге-Куты на відрізку $[a, b]$ з кроком h і початковою умовою $y(a) = y_0$. Початкові дані для виконання завдання беруться з таблиці 14.1. Побудувати графік.

Таблиця 14.1 – Варіанти завдань

№ п/п	Функція	a	b	y_0	h
1	$\frac{3x - y}{x^2 + y}$	2	3	1	0.1
2	$\frac{2x + y + 4}{2y + x}$	3	4	1	0.1
3	$\frac{x^2 - y}{2x + y + 1}$	0	1	2	0.1
4	$\frac{x^2 - y + 2}{xy + 3x}$	2	3	1	0.1
5	$\frac{3 - x - y^2}{2 - xy^2}$	1	2	1	0.1
6	$\frac{2 - x - y^2 x}{3x + y}$	0	1	1	0.1
7	$\frac{1 + 3xy}{5 - x + y^2}$	0	1	2	0.1
8	$\frac{x^2 y + 2}{2x - y}$	0	1	1	0.1

Продовження таблиці 14.1

№ n/n	Функція	a	b	y ₀	h
9	$\frac{x^2 + y + 2}{2x - y}$	2	3	2	0.1
10	$\frac{xy + 4}{2y - xy + 1}$	0	1	3	0.1
11	$y - 4 * x + 3$	0	1	3	0.1
12	$\frac{x - y}{x^2 - 2}$	0	1	1	0.1
13	$4x - 2 + y$	1	2	2	0.1
14	$\frac{x - 4y}{x^2}$	1	2	1	0.1
15	$xy - 2 \frac{x}{y^2}$	1	2	1	0.1

Література

- 1 Чепак, Л.В. Численные методы. Использование MatLab [Текст] / Л.В. Чепак, А.Г. Масловская. – Благовещенск, 2005 г. – 69 с.
- 2 Киреев, В.И. Численные методы в системах и задачах [Текст]: учеб.пособие / В.И. Киреев, В.А. Пантелеев. – М.:Выш.шк., 2008 г. – 480 с.
- 3 Половко, А.М. Matlab для студента [Текст] / А.М. Половко, П.Н. Бутусов. – Спб.: «БХВ-Петербург», 2005 г. – 317 с.

Розробив: Ланська С.С.
 Розглянуто та схвалено
 на засіданні циклової комісії
 програмної інженерії
 Протокол № 4 від 17.11.2021 р.
 Голова комісії _____ Ланська С.С.