МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

„КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ”

КОМП’ЮТЕРНІ ТЕХНОЛОГІЇ ТА ПРОГРАМУВАННЯ-2

ПРОГРАМНІ ЗАСОБИ ЧИСЛОВОГО АНАЛІЗУ

МЕТОДИЧНІ ВКАЗІВКИ

до виконання практикуму для студентів напряму підготовки
„Автоматизація та комп’ютерно-інтегровані технології”

Київ

НТУУ “КПІ”

2014

Комп’ютерні технології-2. Програмні засоби числового аналізу : Метод. вказівки до викон. практикуму для студ. напр. „Автоматизація та комп'ютерно-інтегровані технології” / Уклад.: Д.О. Ковалюк. – К. : НТУУ ”КПІ“, 2013. – 38с.

Гриф надано Вченою радою ІХФ
(Протокол № 2 від 24 лютого 2014 р.)

Навчальне видання

КОМП’ЮТЕРНІ ТЕХНОЛОГІЇ-2

ПРОГРАМНІ ЗАСОБИ ЧИСЛОВОГО АНАЛІЗУ

Методичні вказівки до виконання практикуму для студентів напряму підготовки „Автоматизація та комп'ютерно-інтегровані технології”

Укладач: Ковалюк Дмитро Олександрович, канд. техн. наук, доц.

Відповідальний редактор А.І. Жученко, докт. техн. наук, проф.

Рецензент О.Л. Сокольський, канд. техн. наук, доц.

Авторська редакція

ЗМІСТ

ВСТУП

На сьогодні системи автоматизації математичних розрахунків класу Mathcad залишаються єдиними математичними системами, в яких опис та розв’язання математичних задач задається звичними математичними формулами і знаками. Такий же звичний вигляд мають і результати обчислень.

Потужний інструментарій вбудованих функцій робить систему незамінною при розв’язанні задач аналізу даних – і найбільш доступною користувачеві по використанню.

В методичних вказівках пропонується набути навичок розв’язання основних класів задач аналізу даних в математичному пакеті Mathcad. Особливістю роботи є те, що кожне завдання вимагає виконати власну програмну реалізацію чисельних алгоритмів та порівняти результати з вбудованими функціями Mathcad.

МЕТА І ЗАВДАННЯ ПРАКТИЧНИХ ЗАНЯТЬ

Метою практикуму є формування у студентів компетенцій щодо створення та дослідження математичних моделей об'єктів, а саме: знання сучасних програмних засобів математичного моделювання та числового аналізу; основних конструкцій мов програмування; чисельних алгоритмів розв'язання рівнянь, нерівностей та їх систем; чисельних алгоритмів обробки експериментальних даних; здатність розв'язувати лінійні рівняння та їх системи; здатність отримувати математичні моделі об'єктів з використанням регресійного аналізу, апроксимації та інтерполяції даних.

У результаті вивчення студент повинен знати:

- сучасні програмні засоби числового для числового аналізу.
- основні конструкції мов програмування;
- методи розв'язування систем лінійних алгебраїчних рівнянь;
- методи регресійного аналізу, апроксимації та інтерполяції даних.
- засоби візуалізації результатів розв'язання

Після практичних занять студенти повинні вміти:

- розв'язувати системи рівнянь за допомогою вбудованих та власних програмно-реалізованих функцій;
- виконувати апроксимацію експериментальних даних;
- представляти результати розв'язання у табличному та графічному вигляді;
- аналізувати точність чисельних методів та створених моделей.

ПРАКТИЧНЕ ЗАНЯТТЯ №1

ЗАСОБИ ПРОГРАМУВАННЯ В MATHCAD

Мета роботи – ознайомитися з вбудованою мовою програмування *MathCad*, вивчити основні синтаксичні конструкції.

Теоретичні відомості

Для вставки програмного коду в документи Mathcad використовується спеціальна панель **Programming** (Программирование), яку можна викликати на екран, натиснувши кнопку **Programming Toolbar** на панелі **Math** (Математика).

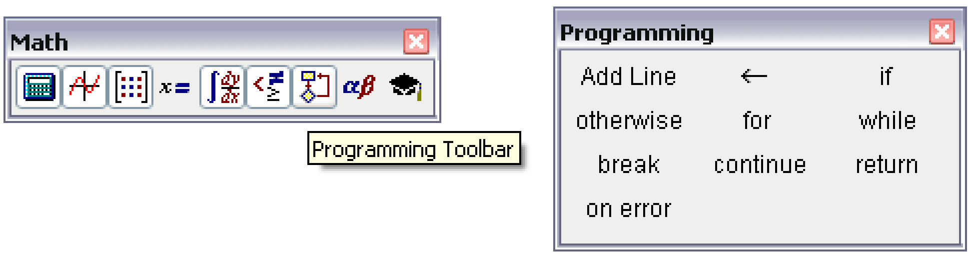


Рис. 1. Панель програмування

Програмування в Mathcad має ряд переваг, які роблять документ більш простим і таким, що легко читається:

1. можливість застосування циклів та умовних операторів;
2. простота створення функцій та змінних;
3. використання локальних змінних та обробка помилок.

Реалізувати алгоритм обчислення в Mathcad можна, використовуючи програми, що містять конструкції подібні до конструкцій мови програмування Pascal – оператори присвоєння, оператори циклів, умовні оператори, тощо.

Mathcad дозволяє оформлювати програмний код у вигляді підпрограм, аналогічно до процедур і функцій в інших мовах програмування.

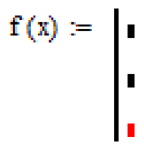
Опис програми, оформленої у вигляді функції, розміщується в робочому документі перед її викликом і містить ім’я функції, список формальних параметрів і тіло функції. Кожна функція MathCad має своє ім’я, використовуючи яке здійснюється її виклик. Після імені в дужках записується список формальних параметрів, через які в функцію передаються реальні дані для виконання обчислень. Якщо функція не має формальних параметрів, тоді дані передаються через імена змінних, визначених в тілі основного документу.

Порядок опису функції MathCad

Для введення в робочий документ функції необхідно:

1. ввести ім’я функції і список формальних параметрів та ввести символ “:”;

2. відкрити панель **Програмування** та клацнути кнопку “**Add line**”.



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

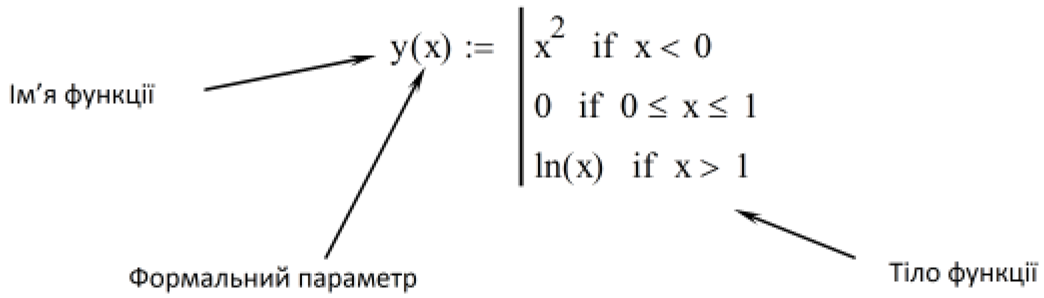
1. Перейти в перше поле і ввести перший оператор тіла програми-функції. Нижнє поле завжди призначене для визначення значень, які повертаються програмою. Для того, щоб ввести додаткові поля для

введення операторів, потрібно натиснути кнопку “**Add line**”. Для видалення того чи іншого оператора або поля введення з тіла програми-функції, потрібно виділити його рамкою і натиснути клавішу **Delete**.

2. Якщо приблизно відомо, скільки рядків коду міститиме програма, можна створити потрібну кількість ліній повторними натисненнями кнопки **Add Line**.

3. Заповнити нижнє поле виразом, який визначає значення, що повертається з функції.

Приклад програми функції:



На одному листі MathCAD можуть використовуватися один або декілька програмних блоків. Звичайно їх використовують при розробці функцій, які здійснюють яку–небудь складну обробку даних, наприклад, знаходять корінь нестандартного рівняння.

Змінні. В програмному блоці можна отримати доступ до значеннь змінних, визначених в MathCAD до цього блоку. Проте змінити значення цих змінних усередині програмного блоку неможливо. Всі змінні, яким присвоюються значення всередині програмного блоку, будуть локальними змінними, які недоступні зовні блоку.

Спеціально задавати змінні не потрібно, достатньо просто присвоїти їм значення. Якщо програмний блок є тілом функції, то він також може читати значення аргументів цієї функції.

Огляд програмних операторів Mathcad:

Команда	**Функція**	**Приклад**
Add Line	Додати новий програмний рядок	
←	Присвоювання значення локальної змінної.	$y \leftarrow 0$
if	Умовний оператор (оператор розгалуження) if; умова повинна стояти після if, а оператор, що виконується, якщо виконано задану умову – перед if.	$f(x) := \left\vert \begin{array}{l} -5 \text{ if } x > 0 \\ 9 \text{ otherwise} \end{array} \right.$ $f(4) = -5 \quad f(-10) = 9$
otherwise	Оператор, що задає альтернативну гілку умовного оператора. Позначає оператор, що повинен бути виконаний, якщо умова оператора if не виконується.	$f(x) := \left\vert \begin{array}{l} -5 \text{ if } x > 0 \\ 9 \text{ otherwise} \end{array} \right.$ $f(4) = -5 \quad f(-10) = 9$
for	Оператор циклу з параметром. За ключовим словом for слідує змінна-лічильник, а після символу приналежності вводиться проміжок зміни цієї змінної.	$suma(n) := \left\vert \begin{array}{l} s \leftarrow 0 \\ \text{for } i \in 1..n \\ \quad s \leftarrow s + i \\ s \end{array} \right.$ $suma(9) = 45$
while	Оператор циклу с передумовою. Внутрішні оператори циклу будуть виконуватися доти, доки буде істинною умова, що слідує за ключовим словом while.	$K(x,f) := \left\vert \begin{array}{l} \text{while } f(x) < 110 \\ \quad x \leftarrow 5 + f(x) \\ x \end{array} \right.$ $K(7, \tan) = 7.845$

break	Опертор дострокового припинення циклу або програми. Служить для передчасного завершення циклу, щоб, уникнути зайвих обчислень.	break if $i \geq 10$
continue	Оператор переходу до наступної ітерації. Служить для передчасного завершення поточної ітерації циклу; сам цикл при цьому триває.	continue if $x > 10$
retutn	Передчасне завершення програми; зазначене в комірці значення буде повернуто.	return y
on error	Оператор, що визначає значення, яке повертається у випадку виникнення помилки. Якщо при обчисленні виразу expr2 виникла помилка, обчислюється вираз expr1.	expr1 on error expr2

Програмний блок є групою операторів присвоєння і керуючих операторів. Необхідно звернути особливу увагу, що всі ключові слова (наприклад, if) в цих операторах обов’язково вводяться за допомогою панелі Programming (Програмування).

В цілому правила роботи з операторами ті ж, що і в мові Pascal, відмінності стосуються способу запису операторів.

Якщо функція є програмним блоком, то значення, яке повертає функція, – це звичайно значення, яке обчислене останнім оператором блоку, що спрацював.

ПРАКТИЧНЕ ЗАНЯТТЯ №2

РОЗВ'ЯЗАННЯ СИСТЕМ ЛІНІЙНИХ АЛГЕБРАЇЧНИХ РІВНЯНЬ МЕТОДОМ ГАУСА З ВИКЛЮЧЕННЯМ

Мета роботи – виконати програмну реалізацію методу Гауса з виключенням для розв'язання систем лінійних алгебраїчних рівнянь.

Теоретичні відомості

Одним з точних методів розв'язання системи лінійних алгебраїчних рівнянь (СЛАР) є метод Гауса з виключенням, що складається з двох етапів [1]:

Прямий хід в результаті якого СЛАР, зводиться до еквівалентної системи з верхньою трикутною матрицею коефіцієнтів виду:

$$\begin{cases} a_{11}\cdot x_1 & +a_{12}\cdot x_2 & +\ldots+ & a_{1n}\cdot x_n & =b_1 \\ 0\cdot x_1 & +a_{22}\cdot x_2 & +\ldots+ & a_{2n}\cdot x_n & =b_2 \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ 0\cdot x_1 & +0\cdot x_2 & +\ldots+ & a_{nn}\cdot x_n & =b_n \end{cases} \qquad (1)$$

Зворотній хід, що дозволяє знайти вектор розв'язків. Починаючи з останнього рівняння знаходяться елементи вектору розв'язків, які використовуються для розрахунку наступного елементу шляхом підстановки в попереднє рівняння системи (1).

Алгоритм метода Гауса з виключенням передбачає послідовне виключення *i-1* невідомих вектора $\bar{x}$ з усіх рівнянь системи, починаючи з другого. Це реалізується за рахунок наступних властивостей, що не змінюють тотожність рівняння:

1) множення обох частин рівняння на будь-яке число, крім нуля;
2) додавання до обох частин рівняння будь-якого числа, крім нуля.

Алгоритм зведення до системи з трикутною матрицею коефіцієнтів [1] покажемо на прикладі наступної системи:

$$\begin{cases} a_{11}\cdot x_1 + a_{12}\cdot x_2 + a_{13}\cdot x_3 = b_1 \\ a_{21}\cdot x_1 + a_{22}\cdot x_2 + a_{23}\cdot x_3 = b_2 \\ a_{31}\cdot x_1 + a_{32}\cdot x_2 + a_{33}\cdot x_3 = b_3 \end{cases} \tag{2}$$

1) Перевіримо, щоб один із коефіцієнтів a_{11}, a_{21}, a_{31} не дорівнював нулю. Якщо, наприклад, $a_{11} = 0$, тоді необхідно переставити рівняння так, щоб коефіцієнт при *x_1* у першому рівнянні не дорівнював нулю.

2) Обчислюється множник:

$$M_2 = \frac{a_{21}}{a_{11}}. \tag{3}$$

3) Перше рівняння системи (2) множиться на M_2 і віднімається від другого рівняння системи. Результат обчислення має вигляд:

$$0\cdot x_1 + (a_{22} - M_2 a_{12})x_2 + (a_{23} - M_2 a_{13})x_3 = b_2 - M_2 b_1 \tag{4}$$

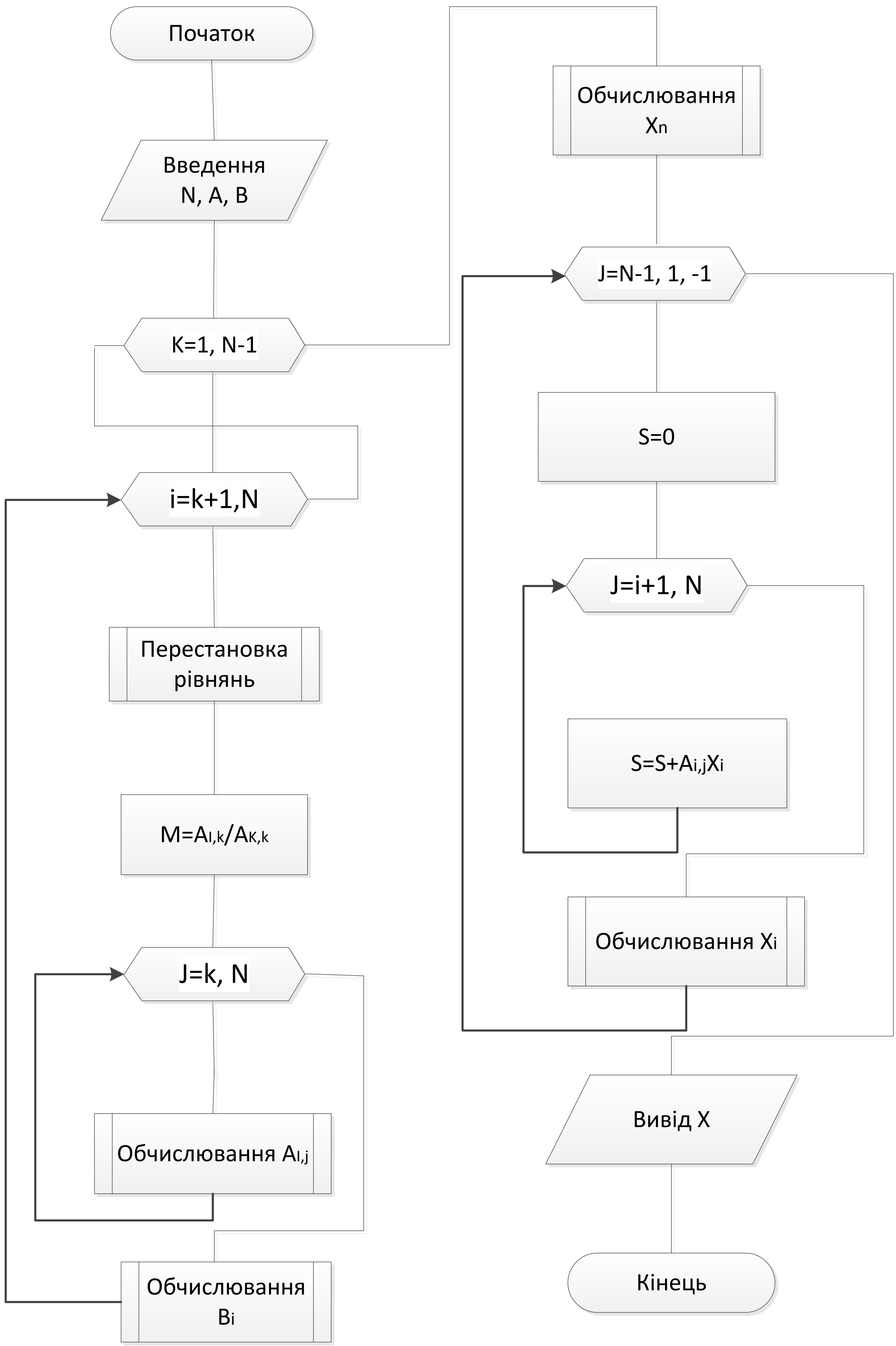
4) Далі необхідно звільнитися від коефіцієнта a_{31} при x_1 в третьому рівнянні системи (2) за аналогічним алгоритмом обчислюється множник для третього рівняння:

$$M_3 = \frac{a_{31}}{a_{11}}. \tag{8}$$

5) Перше рівняння системи (1) множиться на M_3 і віднімається від третього рівняння. Коефіцієнт при x_1 стає нулем, і третє рівняння набуває вигляду:

Аналогічно повторюємо цю операцію для всіх стовпців, від першого до передостаннього.

Алгоритм методу (блок-схема)



Контрольний приклад

$$\begin{cases} 2x_1 + 3x_2 + 3x_3 = 20 \\ 4x_1 + 4x_2 + 2x_3 = 20 \\ 6x_1 + 2x_2 + 4x_3 = 26 \end{cases} \qquad \begin{cases} 2 \cdot x_1 + 3x_2 + 3x_3 = 20 \\ 0 \cdot x_1 - 2x_2 - 4x_3 = -20 \\ 0 \cdot x_1 - 7x_2 - 5x_3 = -34 \end{cases}$$

$$\begin{cases} 2x_1 + 3x_2 + 3x_3 = 20 \\ -2x_2 - 4x_3 = -20 \\ 9x_3 = -36 \end{cases}$$

$X_3 = 4 \qquad X_2 = 2 \qquad X_1 = 1$

Порядок виконання роботи

1. Задати систему рівнянь з наперед відомими коренями та знайти її розв’язок.
2. Задати систему рівнянь, коли вона не має розв’язку.
3. Скласти схему алгоритму розв’язання системи лінійних алгебраїчних рівнянь методом Гауса.
4. Виконати програмну реалізацію алгоритму.

Контрольні запитання

1. В чому суть алгоритмів метода Гауса?
2. Наведіть власний приклад системи рівнянь і розв’яжіть її методом Гауса з послідовним виключенням.
3. В якому випадку система рівнянь немає розв’язків?
4. В чому суть прямого та зворотного ходу в методах Гауса?
5. Які системи називають еквівалентними?

Приклад виконання роботи

ORIGIN := 1

```
gaus(A,B) := | n ← last(B)
             | for  k ∈ 1..n − 1
             |   for  i ∈ k + 1..n
             |     | M ← A[i,k] / A[k,k]
             |     | trace((M))
             |     | for  j ∈ (k)..n
             |     |   | A[i,j] ← A[i,j] − M·A[k,j]
             |     |   | trace(A[i,j])
             |     |   | trace("i={0} j={1} k={2}" ,i,j,k)
             |     | B[i] ← B[i] − M·B[k]
             | for  k ∈ n
             |   X[k] ← 0
             | X[k] ← B[n] / A[n,n]
             | for  i ∈ n − 1..1
             |   | s ← 0
             |   | for  j ∈ i..n
             |   |   s ← s + A[i,j]·X[j]
             |   | X[i] ← (B[i] − s) / A[i,i]
             | X
```

$$A1 := \begin{pmatrix} 2 & 3 & 3 \\ 4 & 4 & 2 \\ 6 & 2 & 4 \end{pmatrix} \qquad B := \begin{pmatrix} 20 \\ 20 \\ 26 \end{pmatrix} \qquad gaus(A1, B) = \begin{pmatrix} \{3,1\} \\ \{3,1\} \\ 4 \end{pmatrix}$$

$$X := A1^{-1} \cdot B \qquad X = \begin{pmatrix} 1 \\ 2 \\ 4 \end{pmatrix} \qquad lsolve(A1, B) = \begin{pmatrix} 1 \\ 2 \\ 4 \end{pmatrix}$$

ПРАКТИЧНЕ ЗАНЯТТЯ №3

РОЗВ'ЯЗАННЯ ЗАДАЧІ ПОШУКУ КОРЕНІВ НЕЛІНІЙНОЇ ФУНКЦІЇ МЕТОДОМ ПОЛОВИННОГО ДІЛЕННЯ

Мета роботи – виконати програмну реалізацію методу половинного ділення для знаходження коренів нелінійної функції.

Постановка задачі

Нехай маємо рівняння $f(x)=0$, де $f(x)$ – неперервна, монотонна нелінійна функція, яка має на відрізку $[a,b]$ єдиний корінь ξ. Потрібно знайти значення кореня ξ з заданою похибкою ε.

Теоретичні відомості

Метод призначений для пошуку коренів нелінійної функції $f(x)$ на відрізку $[a,b]$. На кожній ітерації методу знаходиться середина відрізка – точка c. При цьому корінь може знаходиться на інтервалі [a, c] або [c, b] Необхідною умовою наявності кореня на інтервалі є різні знаки значення функції на кінцях цього інтервалу. Інтервал, на якому відсутній корінь, відкидається. Ділення відрізка навпіл виконується поки довжина інтервалу [a, b] не стане меншою наперед заданої точності методу.

Алгоритм методу (рис.2.1) оснований на багатократному ділені навпіл і звужуванні досліджуваного відрізка $[a,b]$, який отримали в результаті попереднього дослідження функції $f(x)$ (відокремлення коренів).[4]

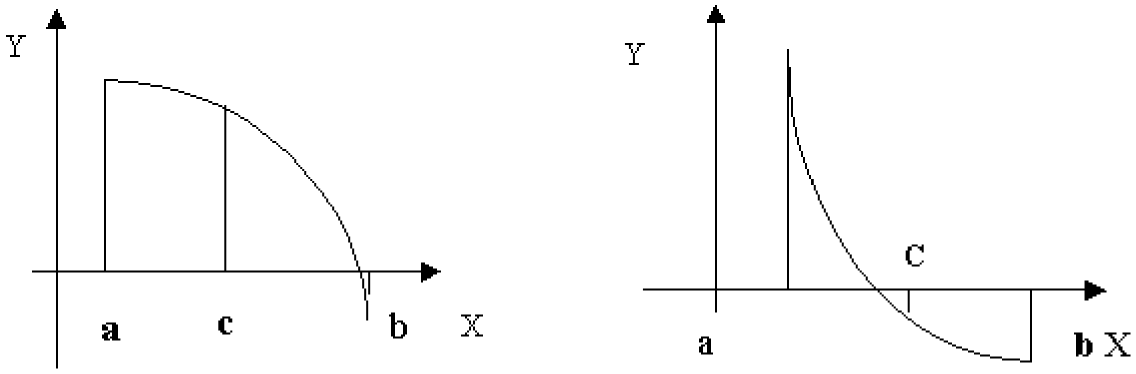


Рис. 2.1. Ділення інтервалу навпіл – графічна інтерпретація

а) корінь на інтервалі [c b]   б) корінь на інтервалі [a c]

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. Закінчуємо роботу алгоритму

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.

3. Процес ділення відрізка навпіл виконується доти, поки на якомусь етапі, або середина відрізка буде коренем, або буде виконана умова закінчення

ітераційного процесу: $|b-a|<\varepsilon$. У цьому випадку за наближене значення кореня вибирають $x=\dfrac{b-a}{2}$.

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

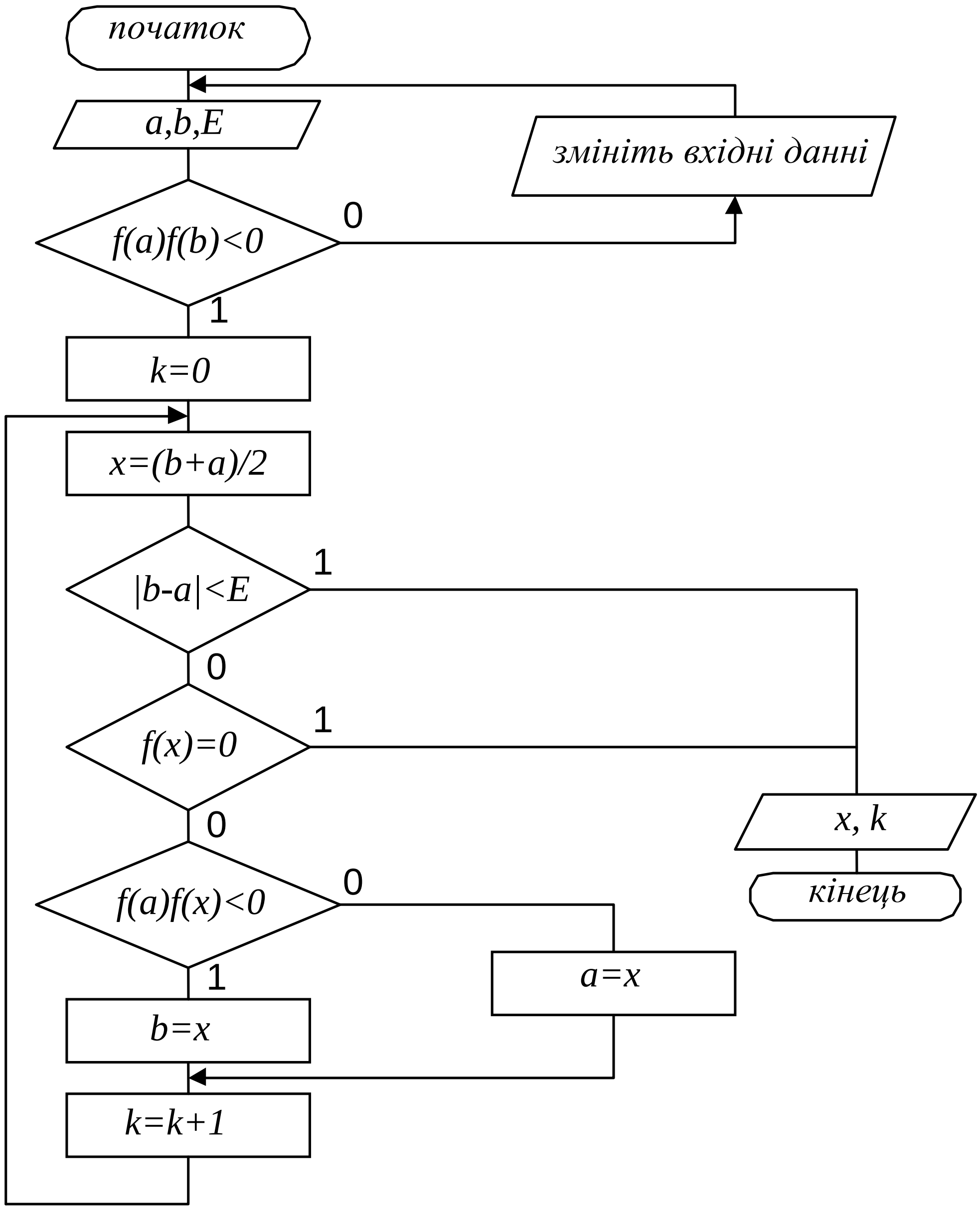


Рис. 2.2 – Схема алгоритму розв'язання нелінійного рівняння методом половинного ділення

Приклад виконання роботи

Знайти корінь функції
f(x)= x^3-2· x^2+2· x-1

[-3; 3] E=0.1

N ітерації	a	c	b	b-a
1	-3	0	3	6
2	0	1,5	3	3
3	0	0,75	1,5	1,5
4	0,75	1,125	1,5	0,75

f(-3)= $(-3)^3$-2· $(-3)^2$+2·(-3)-1=-52

f(0)= $(0)^3$-2· $(0)^2$+2·(0)-1=-1

f(3)= $(3)^3$-2· $(3)^2$+2·(3)-1=14

f(1.5)= $(1.5)^3$-2· $(1.5)^2$+2·(1.5)-1=0.875

f(0.75)= $(0.75)^3$-2· $(0.75)^2$+2·(0.75)-1=-0.203

f(x)= 2· x^2+2· x-2

[-1; 3] E=0.1

N ітерації	a	c	b	b-a
1	-1	1	3	4
2	-1	0	1	2
3	0	0.5	1	1
4	0.5	0.75	1	0.5

f(-1)= 2· $(-1)^2$+2·(-1)-2=-2

f(1)= 2· $(1)^2$+2·(1)-2=2

f(3)= 2· $(3)^2$+2·(3)-2=22

f(0)= 2· $(0)^2$+2·(0)-2=-2

f(0.5)= 2· $(0.5)^2$+2·(0.5)-2=-0.5

Порядок виконання роботи

1. Задати нелінійну функцію, що має корінь на інтервалі, та точність методу
2. Написати програму знаходження кореня.
3. Виконати 2 ітерації знаходження коренів математично, порівняти результати програми та контрольного прикладу

Контрольні запитання

1. В чому суть методу половинного ділення.
2. Умова знаходження кореня функції на інтервалі.
3. Умова закінчення роботи алгоритму.

Приклад виконання роботи

```
polovyn_root(a,b,f,ε) := | iterac ← 0
                         | while |b − a| > ε
                         |   | c ← (a + b)/2
                         |   | Rozvyazok[iterac,0 ← iterac + 1
                         |   | Rozvyazok[iterac,1 ← a
                         |   | Rozvyazok[iterac,2 ← c
                         |   | Rozvyazok[iterac,3 ← b
                         |   | b ← c  if  f(a)·f(c) < 0
                         |   | a ← c  otherwise
                         |   | iterac ← iterac + 1
                         | (c, Rozvyazok)
```

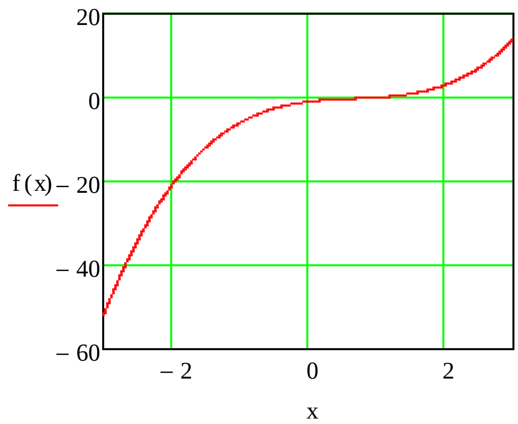
$f(x) := x^3 - 2x^2 + 2x - 1$

r0 := root(f(x), x, −3, 3)

prog_root := polovyn_root(−3, 3, f, 0.1)

r0 = 1

$\text{prog\_root} = \begin{pmatrix} 1.031 \\ \{6,4\} \end{pmatrix}$



$\text{prog\_root}_{0,0} = 1.031$

$$\text{prog\_root}_{1,0} = \begin{pmatrix} 1 & -3 & 0 & 3 \\ 2 & 0 & 1.5 & 3 \\ 3 & 0 & 0.75 & 1.5 \\ 4 & 0.75 & 1.125 & 1.5 \\ 5 & 0.75 & 0.938 & 1.125 \\ 6 & 0.938 & 1.031 & 1.125 \end{pmatrix}$$

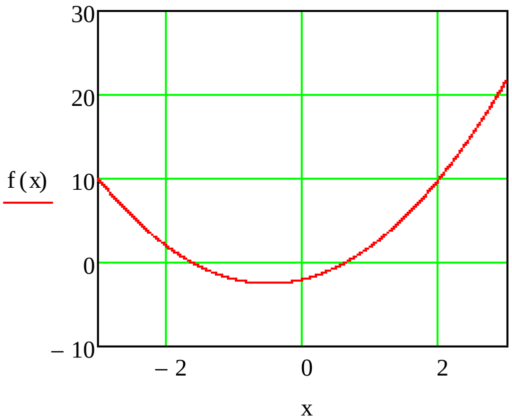
$f(x) := 2x^2 + 2x - 2$

$r0 := \mathrm{root}(f(x), x, -1, 3)$

$\mathrm{prog\_root} := \mathrm{polovyn\_root}(-1, 3, f, 0.1)$

$r0 = 0.618$

$\mathrm{prog\_root} = \begin{pmatrix} 0.563 \\ \{6,4\} \end{pmatrix}$



$\mathrm{prog\_root}_{1,0} = \begin{pmatrix} 1 & -1 & 1 & 3 \\ 2 & -1 & 0 & 1 \\ 3 & 0 & 0.5 & 1 \\ 4 & 0.5 & 0.75 & 1 \\ 5 & 0.5 & 0.625 & 0.75 \\ 6 & 0.5 & 0.563 & 0.625 \end{pmatrix}$

ПРАКТИЧНЕ ЗАНЯТТЯ №4

РОЗВ'ЯЗАННЯ ВИЗНАЧЕНИХ ІНТЕГРАЛІВ

Мета роботи – виконати програмну реалізацію методів прямокутників та трапецій для обчислення визначеного інтегралу, дослідити точність методів.

Теоретичні відомості

Основна ідея чисельного інтегрування полягає в тому, що відрізок інтегрування розбивається на n менших відрізків та визначаються площі фігур, утворені інтегральною функцією на цих відрізках. Значення визначеного інтегралу знаходиться як сума цих площ [1].

Метод прямокутників

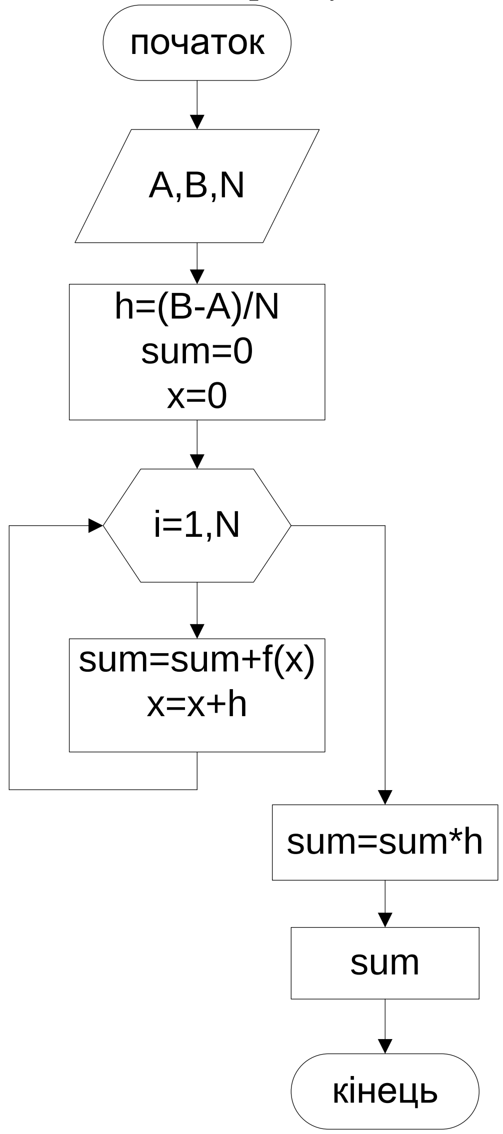
Ідея методу полягає у знаходженні визначеного інтегралу як суми площ N прямокутників з висотою $f(x)$ та основою h, що дорівнює довжині рівних відрізків, на які розбивається початковий відрізок інтегрування $[a, b]$.

$$S_i = h \cdot f(x_i)$$

$$h = \frac{b-a}{N}$$

$$I = h \cdot (f(x_1) + f(x_2) + \ldots + f(x_n)) = h \cdot \sum_{i=1}^{n} f(x_i) = \frac{b-a}{N} \cdot \sum_{i=1}^{n} f(x_i)$$

Блок-схема алгоритму:



A	Початкова точка
B	Кінцева точка
N	Кількість кроків
h	Крок

Програмна реалізація:

```
I_pryamokyt(f, a, b, n) := | h ← (b − a)/n
                           | sum ← 0
                           | x ← a
                           | for  i ∈ 1.. n
                           |     | sum ← sum + f(x)
                           |     | x ← x + h
                           | sum · h
```

Метод трапецій

Ідея методу полягає у знаходженні визначеного інтегралу як суми площ *N* трапецій з основами $f(x_{i-1})$ $f(x_i)$ та висотою h, що дорівнює довжині рівних відрізків, на які розбивається початковий відрізок інтегрування *[a, b]*. Графічно це можна представити наступним чином [1]:

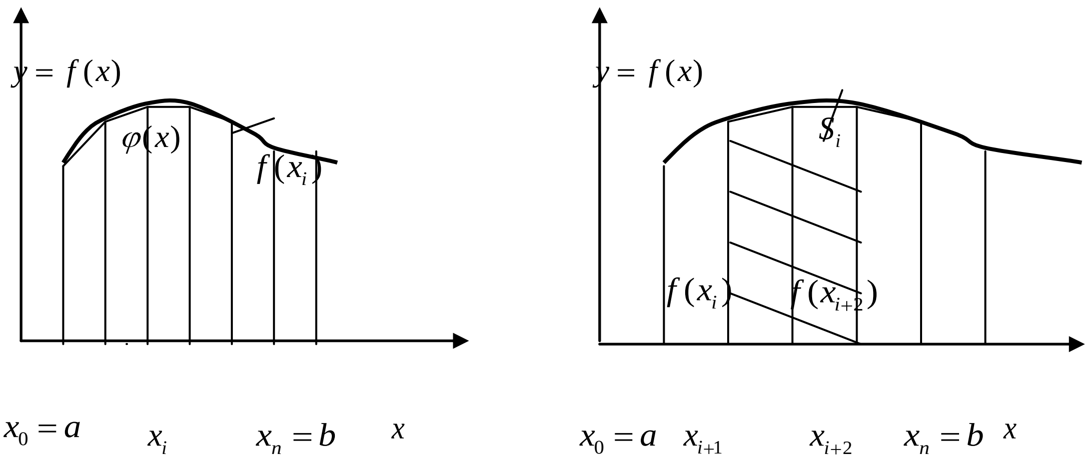


Рис. 4.1.– Геометрична інтерпретація метода трапецій

$$S_i = \frac{f(x_1) + f(x_{i+1})}{2} \cdot h$$

$$S = h \cdot \sum_{i=1}^{N-1} \frac{f(x_1) + f(x_{i+1})}{2}$$

$$h = \frac{b-a}{N}$$

$$S = \frac{b-a}{N} \cdot \left(\frac{y_0}{2} + y_1 + y_2 + \ldots + y_{n-1} + \frac{y_n}{2}\right) = \frac{b-a}{N} \cdot \left(\sum_{i=1}^{N-1} y_i + \frac{y_0 + y_n}{2}\right)$$

Програмна реалізація:

```
I_trapec(f, a, b, n) := | h ← (b − a)/n
                        | sum ← (f(a) + f(b))/2
                        | x ← a
                        | for i ∈ 1.. n − 1
                        |     | sum ← sum + f(x)
                        |     | x ← x + h
                        | sum · h
```

Дослідження точності методів:

$f(x) := 2 \cdot x^2 - 3 \cdot x$

$a := 5$

$b := 10$

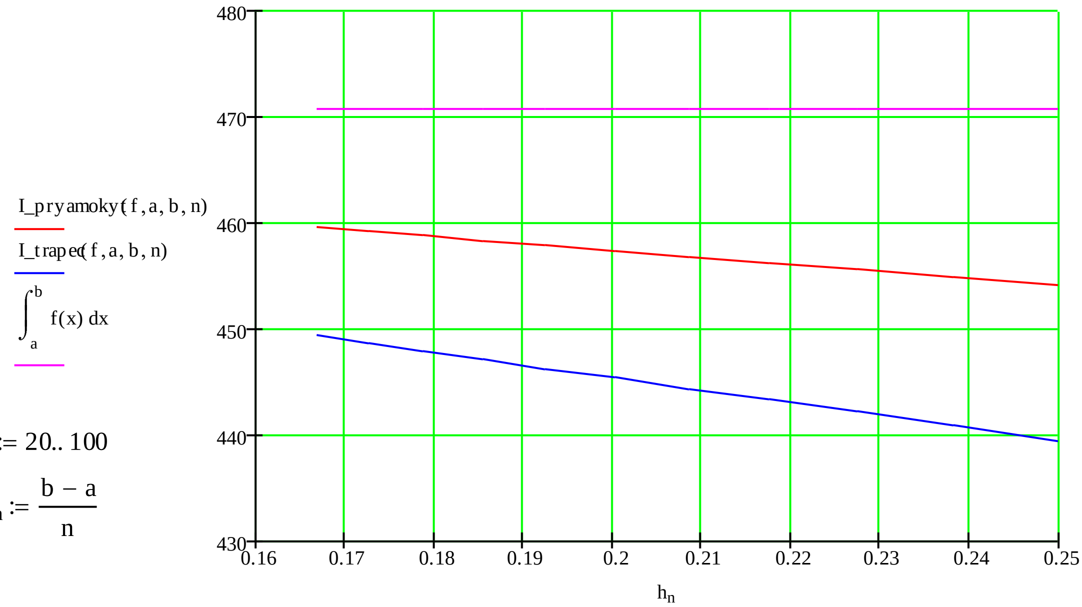
$\int_a^b f(x)\,dx = 470.833$

I_pryamokyt $(f, a, b, 20) = 454.063$

I_trapec $(f, a, b, 20) = 439.469$

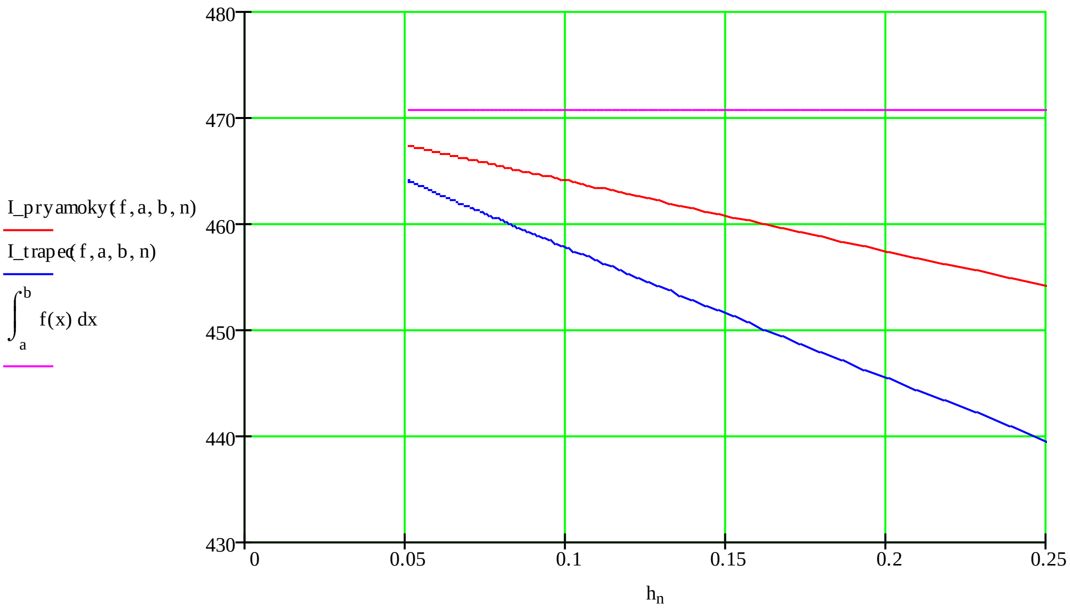
$n := 20..30$

$h_n := \frac{b-a}{n}$



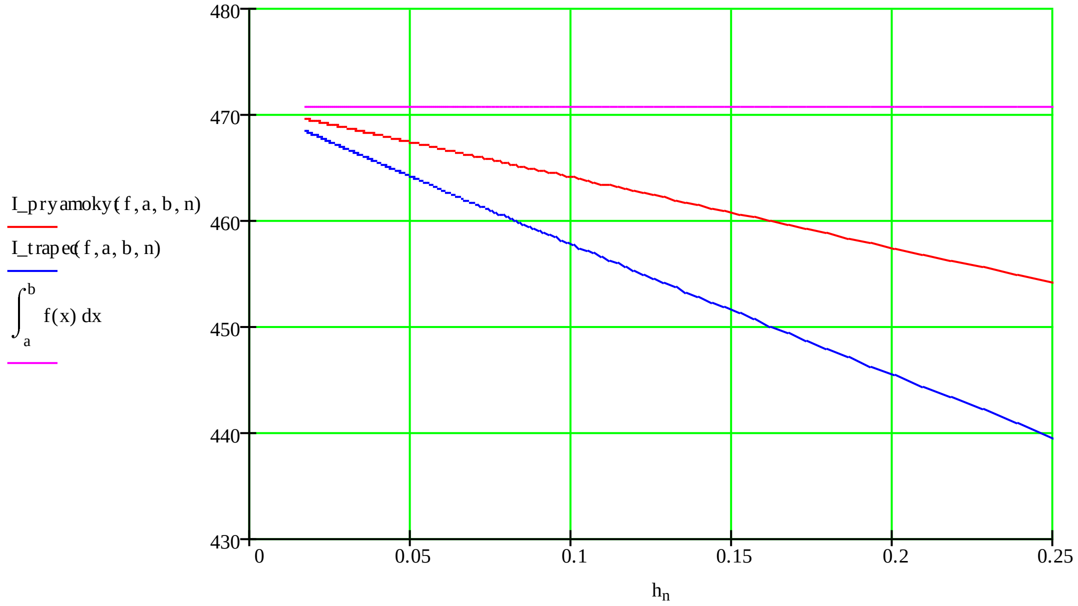
$n := 20..100$

$h_n := \frac{b-a}{n}$



$n := 20..300$

$$h_n := \frac{b-a}{n}$$



Порядок виконання роботи

1. Задати нелінійну функцію, інтервал інтегрування та кількість відрізків.
2. Написати програму обчислення визначеного інтегралу методом прямокутників і трапецій.
3. Розв’язати інтеграл, порівняти точність методів за різної кількості відрізків.

Контрольні запитання

1. В чому суть методів чисельного інтегрування.
2. Формула обчислення за методом прямокутників.
3. Формула обчислення за методом трапецій.

ПРАКТИЧНЕ ЗАНЯТТЯ №5

РЕАЛІЗАЦІЯ ЛІНІЙНОЇ РЕГРЕСІЇ В MATHCAD

Мета роботи — виконати програмну реалізацію побудови лінії регресії в середовищі MathCad та порівняти її з вбудованими функціями.

Постановка задачі

Дано сукупність значень змінної y, що залежать від значень фактору x. Необхідно знайти таку модель, яка б найкраще описувала існуючу залежність.

Теоретичні відомості

Лінійна регресія дозволяє виконати наближення даних (x_i, y_i) у вигляді лінійної функції:

$$y(x) = a + bx, \tag{4.1}$$

де a - зсув по вісі ординат, b - регресійні коефіцієнти.

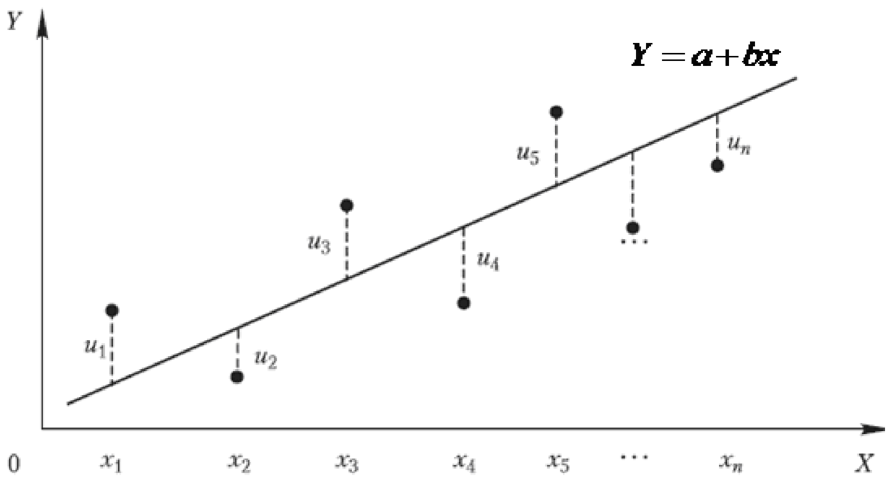


Рис.1. Апроксимація даних лінійною регресією

При знаходженні лінійної регресії основною задачею є визначення параметрів b і a. Лінійну регресію часто називають методом найменших квадратів [5], оскільки коефіцієнти b і a обчислюються з умови мінімізації суми квадратів помилок $|a+bx-y_i|$.

Метод найменших квадратів

Нехай за вибіркою (x_i, y_i), $i = 1, 2, ..., n$, потрібно визначити оцінки a і b рівняння регресії (1) щоб сума квадратів відхилень була мінімальною

$$Y = a + bx$$

У цьому випадку:

$$\sum_{i=1}^{n} u_i^2 = \sum_{i=1}^{n} (\mathrm{y}_i - y)^2 = \sum_{i=1}^{n} (y_i - a - bx_i)^2.$$

Необхідною умовою існування мінімуму функції двох змінних є рівність нулю її частинних похідних:

$$\begin{cases} \dfrac{\partial Q}{\partial a} = -2\sum (y_i - a - bx_i) = 0; \\ \dfrac{\partial Q}{\partial b} = -2\sum (y_i - a - bx_i)x_i = 0; \end{cases}$$

$$\begin{cases} na + b\sum x_i = \sum y_i; \\ a\sum x_i + b\sum x_i^2 = \sum x_i y_i. \end{cases}$$

Поділивши обидва рівняння системи на *n,* отримаємо

$$\begin{cases} a + b\overline{x} = \overline{y}; \\ a\overline{x} + b\overline{x^2} = \overline{xy} \end{cases} \Rightarrow \begin{cases} b = \dfrac{\overline{xy} - \overline{x}\cdot\overline{y}}{\overline{x^2} - \overline{x}^2}; \\ a = \overline{y} - b\overline{x}. \end{cases}, \qquad (4.2)$$

де

$$\overline{x} = \frac{1}{n}\sum x_i, \quad \overline{x^2} = \frac{1}{n}\sum x_i^2, \quad \overline{y} = \frac{1}{n}\sum y_i, \quad \overline{xy} = \frac{1}{n}\sum x_i y_i. \qquad (4.3)$$

Приклад виконання

Дано матрицю значень XY (X-координати в першому стовпці, Y-координати в другому).

line (x, y) — вектор з двох елементів (b, a) коефіцієнтів лінійної регресії a+bx;

intercept (x, y) — коефіцієнт a в лінійній регресії;

slope (x,y) — коефіцієнт b в лінійній регресії.

data :=

0	0
1	2.6
3	23.16
4	27.57
5	24.26
6	16.63
8	30.41
11	47.2
12	50.03
13	60.33
14	59.89
16	71.18
17	84.27
19	77.69

$X := data^{\langle 0 \rangle}$ $\qquad$ $Y := data^{\langle 1 \rangle}$

Кількість точок:

$n := rows(data)$ $\qquad$ $n = 14$

Стандартна похибка: $\qquad SD(x) := stdev(x)\cdot\sqrt{\dfrac{n}{n-1}}$

	X-координати	**Y-координати**
Mean	$\text{mean}(X) = 9.214$	$\text{mean}(Y) = 41.087$
Median	$\text{median}(X) = 9.5$	$\text{median}(Y) = 38.805$
Standard dev.	$SD(X) = 6.192$	$SD(Y) = 27.203$
Variance	$SD(X)^2 = 38.335$	$SD(Y)^2 = 740.008$

Статистика регресії

Intercept	$b_0 := \text{intercept}(X, Y)$	$b_0 = 1.587$
Slope	$b_1 := \text{slope}(X, Y)$	$b_1 = 4.287$
Correlation coeff.	$\text{corr}(X, Y) = 0.9757$	
R^2	$\text{corr}(X, Y)^2 = 0.952$	
Covariance	$\text{cvar}(X, Y) = 152.598$	
Standard Error	$\text{stderr}(X, Y) = 6.204$	
Plots	$r(x) := b_0 + b_1 \cdot x$	

$\text{scale} := \max\left(\overrightarrow{\left|r(X) - Y\right|}\right) \cdot 1.1$

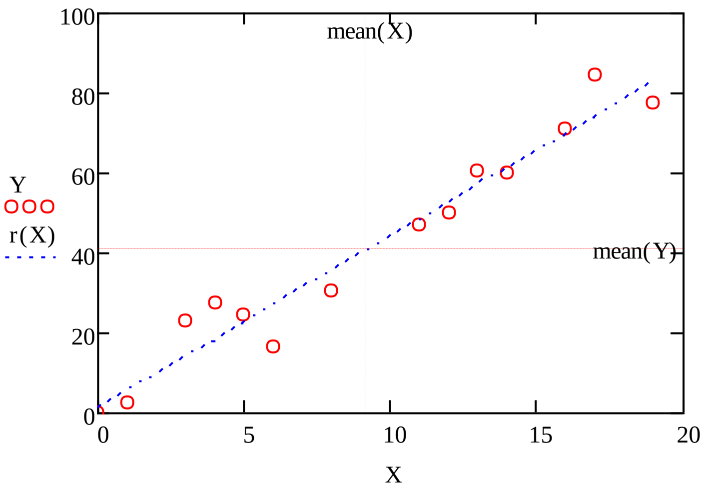


Рис. 3. Графік лінійної регресії

Порядок виконання роботи

1. Задати данні статистичних спостережень в кількості 15 пар
2. Обчислити коефіцієнти регресії за формулою 4.2. та вбудованими функціями
3. Побудувати експериментальні точки та лінію регресії.

Контрольні запитання

1. Ідея методу найменших квадратів.
2. Функції Mathcad для обчислення коефіцієнтів регресії.
3. Точність моделі, обчислення залишків.

ПРАКТИЧНЕ ЗАНЯТТЯ №6

ПРОГРАМУВАННЯ В MATHCAD

Мета роботи – поглибити навики програмування в Mathcad.

Приклад 1. Написати програму знаходження суми позитивних елементів головної діагоналі матриці довільної розмірності.

Змінна	**Що означає**
M	Вхідна матриця елементів
Sum	Сума елементів головної діагоналі матриці
$M_{i,i}$	Елемент головної діагоналі матриці

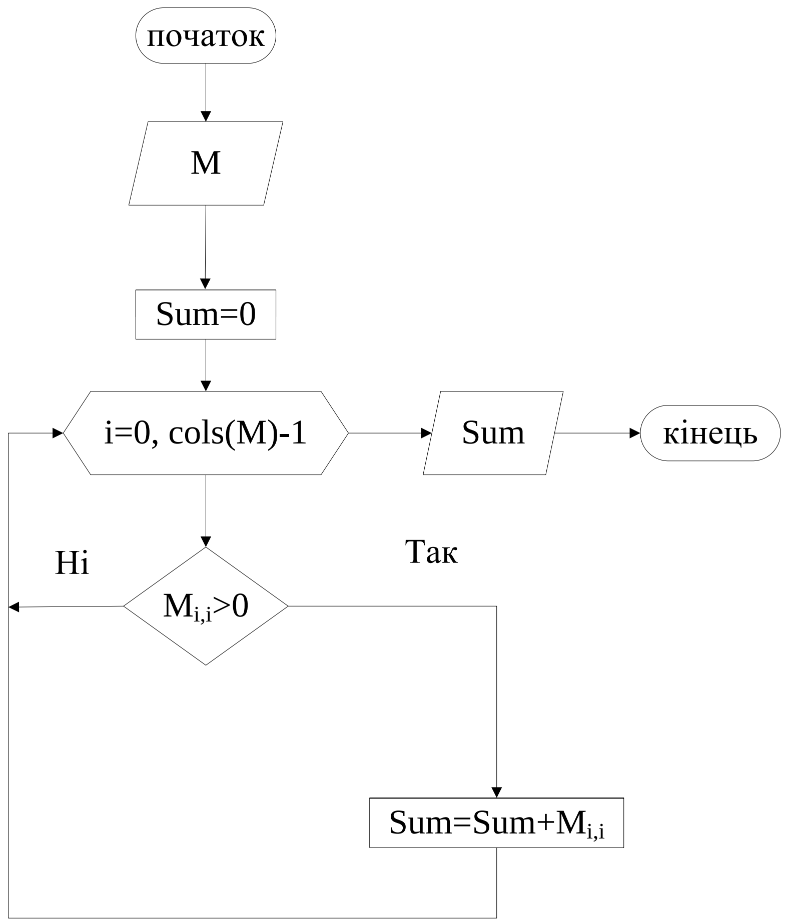


Рис.6.1. Блок-схема алгоритму розв’язання задачі

$$sum(M) := \left| \begin{array}{l} sum \leftarrow 0 \\ \text{for } i \in 0..cols(M)-1 \\ \quad sum \leftarrow sum + M_{i,i} \ \text{ if } M_{i,i} > 0 \\ sum \end{array} \right.$$

$$M := \begin{pmatrix} 4 & 5 & 78 \\ 34 & 2 & 25 \\ 1 & 7 & -3 \end{pmatrix}$$

$$sum(M) = 6$$

Приклад 2. Написати програму знаходження максимального значення функції $y=x^2-2x+3$ на проміжку [-4;10] за умови, що аргумент приймає лише цілі значення.

Змінна	Що означає
A	Вхідна матриця коефіцієнтів функції
p	Початок проміжку, на якому шукаємо максимальне значення
k	Кінець проміжку, на якому шукаємо максимальне значення
current	Поточне значення функції при певному аргументі
next	Наступне значення функції при певному аргументі

$$Maximum(A,p,k) := \left| \begin{array}{l} \text{for } i \in p..k-1 \\ \quad \left| \begin{array}{l} current \leftarrow A_0 \cdot i^2 + A_1 \cdot i + A_2 \\ next \leftarrow A_0 \cdot (i+1)^2 + A_1 \cdot (i+1) + A_2 \\ max \leftarrow current \ \text{ if } current > next \\ max \leftarrow next \ \text{ otherwise} \end{array} \right. \\ max \end{array} \right.$$

$$A := \begin{pmatrix} 1 \\ -2 \\ 3 \end{pmatrix}$$

$$Maximum(A,-4,10) = 83$$

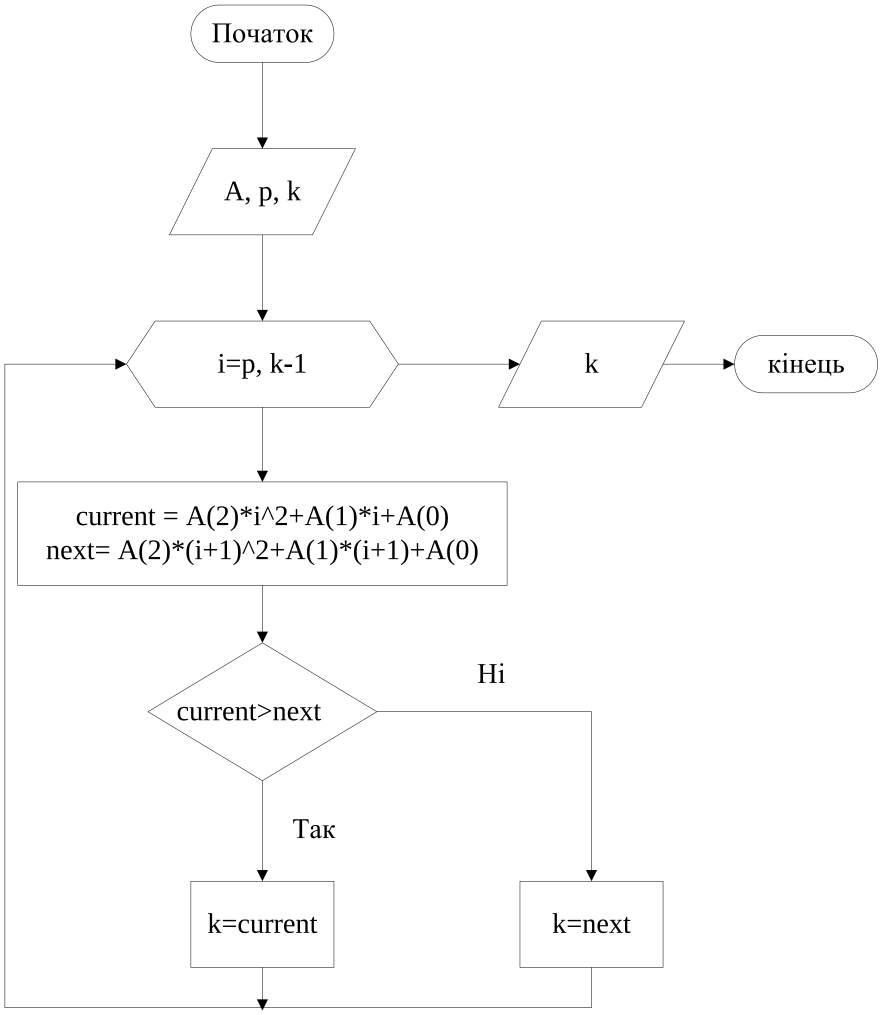


Рис.6.2. Блок-схема алгоритму розв’язання задачі

Приклад 3. Написати програму, яка приймає на вхід вектор цілих чисел, а повертає вектор квадратів відхилень кожного елемента від середнього значення.

Змінна	Що означає
M	Вхідний вектор цілих чисел
Sum	Сума елементів вектора
M_i, V_i	Елемент вектору
ser_znach	Середнє значення
V	Вектор квадратів відхилень кожного елемента від середнього значення

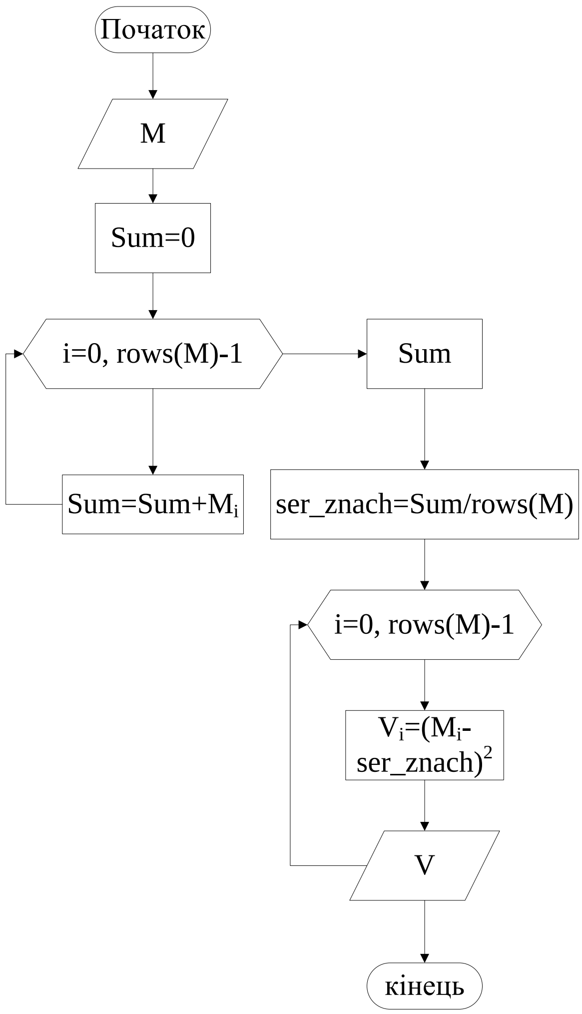


Рис.6.3. Блок-схема алгоритму розв'язання задачі

$$kv\_vidhul(K) := \left| \begin{array}{l} sum \leftarrow 0 \\ \text{for } i \in 0..rows(K) - 1 \\ \quad sum \leftarrow sum + K_i \\ ser\_znach \leftarrow \dfrac{sum}{rows(K)} \\ \text{for } i \in 0..rows(K) - 1 \\ \quad V_i \leftarrow \left| K_i - ser\_znach \right|^2 \\ V \end{array} \right.$$

$$K := \begin{pmatrix} 3 \\ 4 \\ 5 \end{pmatrix}$$

$$kv\_vidhul(K) = \begin{pmatrix} 1 \\ 0 \\ 1 \end{pmatrix}$$

Приклад 4. Написати програму, яка приймає на вхід список студентів групи та прізвище окремого студента, а повертає порядковий номер студента в групі.

Змінна	Що означає
Students	Вхідна матриця зі списком студентів групи
$Students_i$	Певний елемент вхідної матриці
fio	Прізвище певного студента (приймається на вхід)
k	Порядковий номер студента в групі

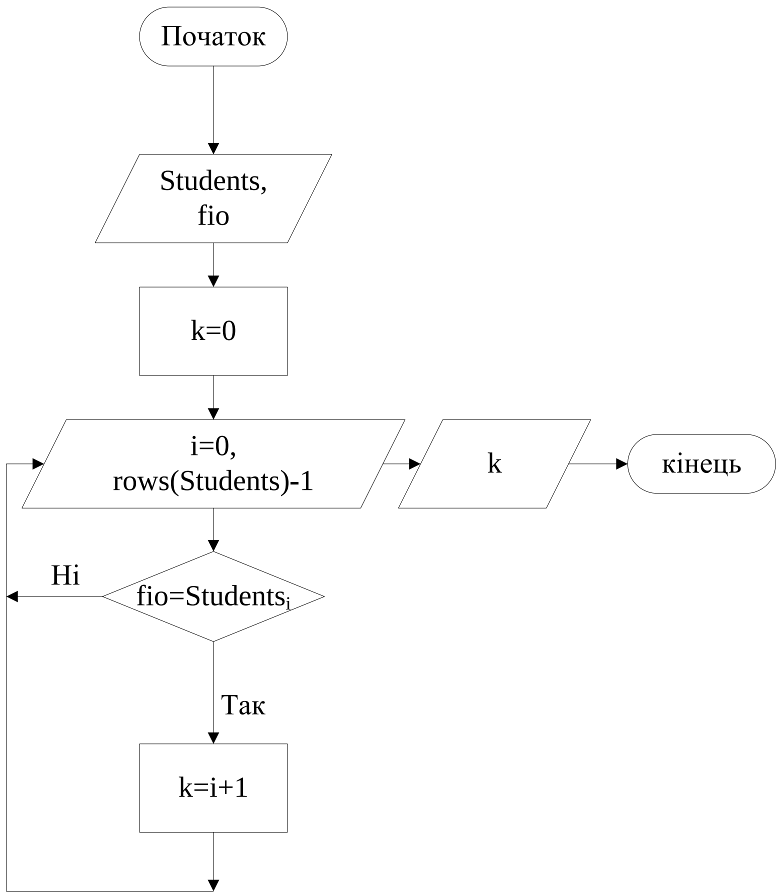


Рис.6.4 . Блок-схема алгоритму розв’язання задачі

Література

1. Москвіна С.М. Числові методи. / С. М. Москвіна - Вінниця, ВНТУ, 2014. - 326 с.

2. Кубрак А. І. Комп’ютерне моделювання та ідентифікація автоматичних систем: Навч. посіб. для студ. напр. "Автоматизація та комп’ютерно-інтегровані технології" / А. І. Кубрак, А. І. Жученко, М. З. Кваско. К. : Політехніка, 2004.- 264с.

3. Жученко А.И. Базовые алгоритмы численого анализа: Учебное пособие. А.И. Жученко, Н.А. Кубрак, И.М. Голинко. – К.: НТУУ «КПИ», 2006. – 236 с.

4. Ладієва Л.Р. Оптимізація технологічних процесів: Навч. посіб. / Київ: Політехніка, 2004.

5. Гурский Д. А., Турбина Е.С. Вычисления в Mathcad 12. – СПб. : Питер, 2006. – 544 с.

6. Макаров Е.Г. Инженерные расчеты в Mathcad. Учебный курс. – СПб. : Питер, 2005. – 448 с.

7. Плис А.И., Сливина Н.А. Mathcad: математический практикум для экономистов и инженеров: Учеб. пособие. – М.: Финансы и статистика, 1999 – 656 с.

8. Кирьянов Д.В. Самоучитель Mathcad.– СПб. : БХВ-Петербург, 2003. – 560 с.