МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ УКРАЇНИ

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ «КПІ»

Кафедра інформаційних систем та технологій

Звіт

з лабораторної роботи № 2

«Розв’язання систем лінійних алгебраїчних рівнянь (СЛАР) прямими методами. Звичайний метод Гауса та метод квадратних коренів»

з дисципліни

«Спеціальні розділи математики-2. Чисельні методи»

Варіант № 23

Перевірила:

доц. Рибачук Людмила Віталіївна

Виконала: Павлова Софія

Студентка гр. ІС-12 , ФІОТ

1 курс,

залікова книжка № ІС-1224

Київ 2022

ВСТУП

Тема: Розв’язання систем лінійних алгебраїчних рівнянь (СЛАР) прямими методами. Звичайний метод Гауса та метод квадратних коренів**.**

Мета: Розв’язати за допомогою Mathcad СЛАР, перевірити розрахунки програмно та розрахувати середньоквадратичну похибку.

Обладнання: Персональні комп’ютери.

ХІД РОБОТИ

Завдання 1:

Розв’язати систему рівнянь з кількістю значущих цифр *n = 6* згідно з варіантом 23 індивідуального завдання.

Вивести всі проміжні результати (*матриці A*, що отримані в ході прямого ходу методу Гауса, матрицю зворотного ходу методу Гауса, або *матрицю T* та *вектор y* для методу квадратних коренів), та розв’язок системи.

Навести результат перевірки: вектор нев’язки ***r = b – Ax***, де *x* – отриманий розв’язок.

Розв’язати задану систему рівнянь за допомогою програмного забезпечення Mathcad.

Навести результат перевірки: вектор нев’язки ***r = b – Axm***, де *xm* – отриманий у Mathcad розв’язок.

Порівняти корені рівнянь, отримані у Mathcad, із власними результатами за допомогою методу середньоквадратичної похибки

$$\delta = \sqrt{\tfrac{1}{n}\sum_{k=1}^{n}(x_k - x_{mk})^2}\ ,$$

Варіант 23:

$\begin{pmatrix} 5{,}18+\alpha & 1{,}12 & 0{,}95 & 1{,}32 & 0{,}83 \\ 1{,}12 & 4{,}28-\alpha & 2{,}12 & 0{,}57 & 0{,}91 \\ 0{,}95 & 2{,}12 & 6{,}13+\alpha & 1{,}29 & 1{,}57 \\ 1{,}32 & 0{,}57 & 1{,}29 & 4{,}57-\alpha & 1{,}25 \\ 0{,}83 & 0{,}91 & 1{,}57 & 1{,}25 & 5{,}21+\alpha \end{pmatrix}$	$\begin{pmatrix} 6{,}19+\beta \\ 3{,}21 \\ 4{,}28-\beta \\ 6{,}25 \\ 4{,}95+\beta \end{pmatrix}$
$\alpha = 0{,}25k,\ k = \lvert Nºвар - 25 \rvert$	$\beta = 0{,}35k,\ k = Nºвар - 21$

Скрін з Mathcad:

Лабораторна робота №2
Варіант №23
Виконала: Павлова Софія, ІС-12

1. Позначимо матрицю системи А:

$k := 2$

$a := 0.25 \cdot k \qquad a = 0.5$

$$A := \begin{bmatrix} 5.18+a & 1.12 & 0.95 & 1.32 & 0.83 \\ 1.12 & 4.28-a & 2.12 & 0.57 & 0.91 \\ 0.95 & 2.12 & 6.13+a & 1.29 & 1.57 \\ 1.32 & 0.57 & 1.29 & 4.57-a & 1.25 \\ 0.83 & 0.91 & 1.57 & 1.25 & 5.21+a \end{bmatrix} \qquad A = \begin{bmatrix} 5.68 & 1.12 & 0.95 & 1.32 & 0.83 \\ 1.12 & 3.78 & 2.12 & 0.57 & 0.91 \\ 0.95 & 2.12 & 6.63 & 1.29 & 1.57 \\ 1.32 & 0.57 & 1.29 & 4.07 & 1.25 \\ 0.83 & 0.91 & 1.57 & 1.25 & 5.71 \end{bmatrix}$$

2. Позначимо вектор правої частини В:

$b := 0.35 \cdot k \qquad b = 0.7$

$$B := \begin{bmatrix} 6.19+b \\ 3.21 \\ 4.28-b \\ 6.25 \\ 4.95+b \end{bmatrix} \qquad B = \begin{bmatrix} 6.89 \\ 3.21 \\ 3.58 \\ 6.25 \\ 5.65 \end{bmatrix}$$

3. Знайдемо розв'язок СЛАР за допомогою вбудованої функції *lsolve(M, v)*:
, де M - матриця дійсних чисел
, а v - вектор дійсних чисел.

$$\mathrm{X} := \mathrm{lsolve}(A, B) = \begin{bmatrix} 0.824 \\ 0.317 \\ -0.025 \\ 1.049 \\ 0.596 \end{bmatrix} \qquad \mathrm{X} = \begin{bmatrix} 0.823737 \\ 0.31733 \\ -0.024814 \\ 1.048714 \\ 0.596426 \end{bmatrix}$$

4. Перевіримо правильність розрахунків. Підставивши початкові значення матриць у рівняння *A*X=B*, маємо одержати правильну рівність:

$$A \cdot \mathrm{X} = \begin{bmatrix} 6.89 \\ 3.21 \\ 3.58 \\ 6.25 \\ 5.65 \end{bmatrix} \qquad B = \begin{bmatrix} 6.89 \\ 3.21 \\ 3.58 \\ 6.25 \\ 5.65 \end{bmatrix}$$

5. Створімо вектор нев'язки *R=B-A*X*:

$$R := B - A \cdot X$$

$$R = \begin{bmatrix} 0 \\ 8.881784 \cdot 10^{-16} \\ 4.440892 \cdot 10^{-16} \\ 8.881784 \cdot 10^{-16} \\ 8.881784 \cdot 10^{-16} \end{bmatrix}$$

Рис. 1. Результат виконання завдання 1 у Mathcad

Код:

У даній лабораторній роботі програмно реалізовано метод Гауса, в основі якого лежить ідея послідовного виключення невідомих, що приводить вихідну систему до трикутного виду, у якому всі коефіцієнти нижче головної діагоналі дорівнюють нулю. У результаті виконання програми отримано верхню трикутну матрицю вигляду:

$$C = \begin{pmatrix} 1 & c_{12} & \dots & c_{1,m-1} & c_{1m} \\ 0 & 1 & \dots & c_{2,m-1} & c_{2m} \\ \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & 1 & c_{m-1,m} \\ 0 & 0 & \dots & 0 & 1 \end{pmatrix}.$$

```
#define _CRT_SECURE_NO_WARNINGS
#include <stdio.h>
#include <stdlib.h>
#include <windows.h>
#include <math.h>
#include <iostream>
using namespace std;

#define ICHAR 80  // Довжина рядку опису системи
#define IDEBUG 1 // Чи друкувати кроки зведення матриці до трикутного вигляду

int matrix_print_off(int nr, int nc, double** A) {
	int i, j;
	if (nr <= 0) return (-1);
	if (nc <= 0) return (-2);
	for (i = 1; i <= nr; i++) {
		for (j = 1; j <= nc; j++) {
			printf("%9.2f  ", A[i][j]);
		}
```

```
		printf("\n");
	}
	return (0);
}

int vector_print_off(int nr, double* x) {
	int i;

	if (nr <= 0) return (-1);

	for (i = 1; i <= nr; i++) {
		cout << "     "<< x[i] << "\n";
	}
	printf("\n");
	return (0);
}

double* matrix_on_vector_mult(int nr, int nc, double** A, double* x) {
	if (nr <= 0) return NULL;
	if (nc <= 0) return NULL;
	double* b = (double*)calloc(nr, sizeof(double)); b--;
	for (int i = 1; i <= nr; i++) {
		b[i] = 0;
		for (int j = 1; j <= nc; j++) b[i] += A[i][j] * x[j];
	}
	return b;
}

void gauss(double** a, double* b, double* x, int n) {
	int   i, j, k, m, rowx;
	double xfac, temp, temp1, amax;

	rowx = 0;
	for (k = 1; k <= n - 1; ++k) {
		amax = (double)fabs(a[k][k]);
		m = k;
		for (i = k + 1; i <= n; i++) {
			xfac = (double)fabs(a[i][k]);
			if (xfac > amax) { amax = xfac; m = i; }
		}
		if (m != k) {
			rowx = rowx + 1;
			temp1 = b[k];
			b[k] = b[m];
			b[m] = temp1;
			for (j = k; j <= n; j++) {
				temp = a[k][j];
```

```
				a[k][j] = a[m][j];
				a[m][j] = temp;
			}
		}
		for (i = k + 1; i <= n; ++i) {
			xfac = a[i][k] / a[k][k];

			for (j = k + 1; j <= n; ++j) {
				a[i][j] = a[i][j] - xfac * a[k][j];
			}
			b[i] = b[i] - xfac * b[k];
		}

		if (IDEBUG == 1) {
			printf("Крок % d:\n\n", k);
			matrix_print_off(n, n, a);
			cout << "\n";
		}
	}

	for (j = 1; j <= n; ++j) {
		k = n - j + 1;
		x[k] = b[k];
		for (i = k + 1; i <= n; ++i) {
			x[k] = x[k] - a[k][i] * x[i];
		}
		x[k] = x[k] / a[k][k];
	}
}

int main(void) {
	SetConsoleCP(1251);
	SetConsoleOutputCP(1251);

	double** a, ** a0, * b, * b0, * x, * xm, * r;
	float  aij, bi, xmi;
	char   desc[ICHAR];
	int    i, j, n;
	FILE* finput;

	finput = fopen("Gaus.TXT", "r");
	if (finput == NULL) {
		printf("Текстовий файл \"Gaus.TXT\" НЕ знайдено!\n");
		return(-1);
	}

	fgets(desc, ICHAR, finput);
```

```
	// Відсканувати перший рядок файлу до 80 знаків
	fscanf(finput, "%d", &n);
	printf("Розмір матриці (N*N) = %d\n\n", n);

	// Виділення пам'яті для матриць (двомірні масиви)
	a = (double**)calloc(n, sizeof(double*)); --a;		// a-- для нумерування 
елементів масивів з одиниці
	a0 = (double**)calloc(n, sizeof(double*)); --a0;	// Копія матриці для перевірки 
розв'язку

	for (i = 1; i <= n; ++i) {
		a[i] = (double*)calloc(n, sizeof(double)); --a[i];
		a0[i] = (double*)calloc(n, sizeof(double)); --a0[i];
	}

	// Виділення пам'яті для векторів (одномірні масиви)
	b = (double*)calloc(n, sizeof(double)); --b;
	b0 = (double*)calloc(n, sizeof(double)); --b0;		// Копія вектора для 
перевірки розв'язку
	x = (double*)calloc(n, sizeof(double)); --x;
	xm = (double*)calloc(n, sizeof(double)); --xm;
	r = (double*)calloc(n, sizeof(double)); --r;

	// Зчитування матриці А
	for (i = 1; i <= n; i++) {
		for (j = 1; j <= n; j++) {
			fscanf(finput, "%f ", &aij);
			a[i][j] = (double)aij;
			a0[i][j] = (double)aij;
		}
	}
	// Зчитування вектора В
	for (i = 1; i <= n; i++) {
		fscanf(finput, "%f ", &bi);
		b[i] = (double)bi;
		b0[i] = (double)bi;
	}
	// Зчитування вектора XM - розв'язку з mathcad
	for (i = 1; i <= n; i++) {
		fscanf(finput, "%f ", &xmi);
		xm[i] = (double)xmi;
	}

	fclose(finput);

	printf("---------------------ВХІДНІ ДАНІ---------------------\n");
```

```
	printf("\nМАТРИЦЯ A:\n\n");
	matrix_print_off(n, n, a);

	printf("\nВЕКТОР B:\n\n");
	vector_print_off(n, b);

	cout << "-------------------------ХІД-------------------------\n\n";
	gauss(a, b, x, n);

	printf("\n----------------------РОЗВ'ЯЗОК----------------------\n\nВЕКТОР X:\n\n");
	vector_print_off(n, x);

	if (IDEBUG == 1) {
		// Створення матричного добутку A*X
		double* a0x = matrix_on_vector_mult(n, n, a0, x);
		printf("----------------------ПЕРЕВІРКА----------------------\n\nВЕКТОР-
ДОБУТОК A*X:\n\n");
		vector_print_off(n, a0x);

		// Створення вектору нев'язки R
		cout << "\nВЕКТОР НЕВ'ЯЗКИ R=B-A*X:\n\n";
		for (int i = 0; i < n; i++) {
			for (int j = 0; j < n; j++) {
				r[i] = b0[i] - a0x[i];
			}
		}
		vector_print_off(n, r);

		// Розрахунок середньоквадратичної похибки
		double err = 0;
		for (i = 1; i <= n; i++) {
			err += pow(x[i] - xm[i], 2);
		}
		err = sqrt(err / n);

		cout << "\n-------------СЕРЕДНЬОКВАДРАТИЧНА ПОХИБКА-------------
\n\nQ  = " << err << "\n\n";
	}

	getchar();
	return(0);
}
```

<u>*Скріншоти виконання програми:*</u>

Microsoft Visual Studio Debug Console

```
Розмір матриці (N*N) = 5

---------------------ВХІДНІ ДАНІ---------------------

МАТРИЦЯ А:

     5.68       1.12       0.95       1.32       0.83
     1.12       3.78       2.12       0.57       0.91
     0.95       2.12       6.63       1.29       1.57
     1.32       0.57       1.29       4.07       1.25
     0.83       0.91       1.57       1.25       5.71

ВЕКТОР В:

     6.89
     3.21
     3.58
     6.25
     5.65

-------------------------ХІД-------------------------

Крок 1:

   1.0000     0.1972     0.1673     0.2324     0.1461
   0.0000     3.1778     1.7256     0.2765     0.6664
   0.0000     2.0344     6.8117     1.1255     1.5065
   0.0000     0.2346     0.8100     2.8509     0.8008
   0.0000     0.8992     1.7243     1.2736     6.7334

Крок 2:

   1.0000     0.1972     0.1673     0.2324     0.1461
   0.0000     1.0000     0.5430     0.0870     0.2097
   0.0000     0.0000     2.8052     0.4662     0.5308
   0.0000     0.0000     2.9092    12.0635     3.2034
   0.0000     0.0000     1.3746     1.3294     7.2785

Крок 3:

   1.0000     0.1972     0.1673     0.2324     0.1461
   0.0000     1.0000     0.5430     0.0870     0.2097
   0.0000     0.0000     1.0000     4.1466     1.1011
   0.0000     0.0000     0.0000    -3.9804    -0.9119
   0.0000     0.0000     0.0000    -3.1795     4.1939

Крок 4:

   1.0000     0.1972     0.1673     0.2324     0.1461
   0.0000     1.0000     0.5430     0.0870     0.2097
   0.0000     0.0000     1.0000     4.1466     1.1011
  -0.0000    -0.0000    -0.0000     1.0000     0.2291
   0.0000     0.0000     0.0000     0.0000    -1.5481

Крок 5:

   1.0000     0.1972     0.1673     0.2324     0.1461
   0.0000     1.0000     0.5430     0.0870     0.2097
   0.0000     0.0000     1.0000     4.1466     1.1011
  -0.0000    -0.0000    -0.0000     1.0000     0.2291
  -0.0000    -0.0000    -0.0000    -0.0000     1.0000
```

```
----------------------РОЗВ'ЯЗОК----------------------

ВЕКТОР X:

     0.823737
     0.31733
     -0.0248144
     1.04871
     0.596426

----------------------ПЕРЕВІРКА----------------------

ВЕКТОР-ДОБУТОК A*X:

     6.89
     3.21
     3.58
     6.25
     5.65


ВЕКТОР НЕВ'ЯЗКИ R=B-A*X:

     0
     0
     4.44089e-16
     0
     0


-------------СЕРЕДНЬОКВАДРАТИЧНА ПОХИБКА--------------

Q  = 0.000308393
```

Рис. 2. Результат виконання завдання 1 програмно

ВИСНОВОК

У ході виконання лабораторної роботи я дізналася про алгоритми розв’язання СЛАР, а саме метод квадратного кореня та різні варіації методу Гауса (з вибором головного елементу, метод прогону). Дізналася про інструменти роботи з матрицями у програмі Mathcad. Я навчилась програмно реалізовувати метод Гауса, у результаті якого було отримано верхню трикутну матрицю та розв’язки системи рівнянь і навчилась використовувати вищезазначений метод у середовищі Mathcad для розв’язання представлених СЛАР.

У результаті виконання програми за допомогою середньоквадратичної похибки було порівняно результати виконання методу Гауса в Mathcad та програмно. Отримана похибка є допустимою, а отже всі розрахунки було зробрено правильно.