

**Міністерство освіти і науки України
Національний технічний університет України
"Київський політехнічний інститут імені Ігоря Сікорського"
Фізико-технічний інститут**

«Методи обчислень»

Лабораторна робота №2
Варіант 2

«Розв’язання систем лінійних алгебраїчних рівнянь (СЛАР) прямими методами»

Виконала:
студентка групи ФБ-95
Гурджия Валерія Вахтангівна

Київ 2022

Завдання

Розв'язати систему рівнянь з кількістю значущих цифр $m = 6$. Якщо матриця системи симетрична, то розв'язання проводити за методом квадратних коренів, якщо матриця системи несиметрична, то використати метод Гауса. Вивести всі проміжні результати (матриці A , що отримуються в ході прямого ходу методу Гауса, матрицю зворотного ходу методу Гауса, або матрицю T та вектор u для методу квадратних коренів), та розв'язок системи. Навести результат перевірки: вектор нев'язки $r = |b - Ax|$, де x - отриманий розв'язок.

№ вар.	Матриця системи A	Вектор правої частини b
2	1,00 0,42 0,54 0,66 0,42 1,00 0,32 0,44 0,54 0,32 1,00 0,22 0,66 0,44 0,22 1,00	0,3 0,5 0,7 0,9

Матриця системи симетрична, тому розв'язання проводимо за методом квадратних коренів.

1) Прямий хід: факторизація $A = TT'$, де

$$A = \begin{pmatrix} 1,00 & 0,42 & 0,54 & 0,66 \\ 0,42 & 1,00 & 0,32 & 0,44 \\ 0,54 & 0,32 & 1,00 & 0,22 \\ 0,66 & 0,44 & 0,22 & 1,00 \end{pmatrix} \quad T = \begin{pmatrix} t_{11} & 0 & \dots & 0 \\ t_{21} & t_{22} & \dots & 0 \\ \dots & \dots & \dots & \dots \\ t_{n1} & t_{n2} & \dots & t_{nn} \end{pmatrix}$$

1.1) Знаходимо елементи t_{ij} матриці T :

$$t_{11} = \sqrt{a_{11}} = \sqrt{1} = 1$$

$$t_{1j} = \frac{a_{1j}}{t_{11}} \quad (i > 1)$$

$$t_{21} = \frac{a_{21}}{t_{11}} = \frac{a_{21}}{1} = 0,42$$

$$t_{31} = \frac{a_{31}}{t_{11}} = \frac{a_{31}}{1} = 0,54$$

$$t_{41} = \frac{a_{41}}{t_{11}} = \frac{a_{41}}{1} = 0,66$$

$$t_{ii} = \sqrt{a_{ii} - \sum_{k=1}^{i-1} t_{ik}^2} \quad (1 < i \leq n)$$

$$t_{ij} = \frac{a_{ij} - \sum_{k=1}^{j-1} t_{ik}t_{jk}}{t_{jj}} \quad (i > j)$$

$$t_{22} = \sqrt{a_{22} - \sum_{k=1}^1 t_{2k}^2} = \sqrt{1 - t_{21}^2} = \sqrt{1 - 0,42^2} = 0,907524$$

$$t_{32} = \frac{a_{32} - \sum_{k=1}^1 t_{3k}t_{2k}}{t_{22}} = \frac{0,32 - 0,54 * 0,42}{0,907524} = 0,102697$$

$$t_{42} = \frac{a_{42} - \sum_{k=1}^1 t_{4k}t_{2k}}{t_{22}} = \frac{0,44 - 0,66 * 0,42}{0,907524} = 0,179389$$

$$t_{33} = \sqrt{a_{33} - \sum_{k=1}^2 t_{3k}^2} = \sqrt{1 - (t_{31}^2 + t_{32}^2)} = \sqrt{1 - (0,54^2 + 0,102697^2)} = 0,835376$$

$$t_{43} = \frac{a_{42} - \sum_{k=1}^2 t_{4k}t_{3k}}{t_{33}} = \frac{0,22 - (0,66 * 0,54 + 0,179389 * 0,102697)}{0,835376} = -0,18533$$

$$t_{44} = \sqrt{a_{44} - \sum_{k=1}^3 t_{4k}^2} = \sqrt{1 - (t_{41}^2 + t_{42}^2 + t_{43}^2)} = \sqrt{1 - (0,66^2 + 0,179389^2 + (-0,18533)^2)} \\ = 0,7056$$

$$t_{ij} = 0 \ (i < j)$$

$$T = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0,42 & 0,907524 & 0 & 0 \\ 0,54 & 0,102697 & 0,835376 & 0 \\ 0,66 & 0,179389 & -0,18533 & 0,7056 \end{pmatrix}$$

$$T' = \begin{pmatrix} 1 & 0,42 & 0,54 & 0,66 \\ 0 & 0,907524 & 0,102697 & 0,179389 \\ 0 & 0 & 0,835376 & -0,18533 \\ 0 & 0 & 0 & 0,7056 \end{pmatrix}$$

1.2) Замість вихідної системи розв'язуємо дві наступні системи:

$$Ty = b, T'x = y.$$

Оскільки відповідні матриці трикутні, одразу можна виконати зворотний хід.

2) Зворотний хід.

2.1) Послідовно знаходимо:

$$y_1 = \frac{b_1}{t_{11}} = \frac{0,3}{1} = 0,3$$

$$y_i = \frac{b_i - \sum_{k=1}^{i-1} t_{ik}y_k}{t_{ii}} \ (i > 1)$$

$$y_2 = \frac{b_2 - \sum_{k=1}^1 t_{2k}y_k}{t_{22}} = \frac{b_2 - t_{21}y_1}{t_{22}} = \frac{0,5 - 0,42 * 0,3}{0,907524} = 0,41211$$

$$y_3 = \frac{b_3 - \sum_{k=1}^2 t_{3k}y_k}{t_{33}} = \frac{b_3 - (t_{31}y_1 + t_{32}y_2)}{t_{33}} = \frac{0,7 - (0,42 * 0,3 + 0,102697 * 0,41211)}{0,835376} \\ = 0,593358$$

$$y_4 = \frac{b_4 - \sum_{k=1}^3 t_{4k}y_k}{t_{44}} = \frac{b_4 - (t_{41}y_1 + t_{42}y_2 + t_{43}y_3)}{t_{44}} \\ = \frac{0,9 - (0,66 * 0,3 + 0,179389 * 0,41211 + (-0,18533) * 0,593358)}{0,7056} = 1,04598$$

$$\begin{cases} y_1 = 0,3 \\ y_2 = 0,41211 \\ y_3 = 0,593358 \\ y_4 = 1,04598 \end{cases}$$

$$x_n = \frac{y_n}{t_{nn}} = x_4 = \frac{y_4}{t_{44}} = \frac{1,04598}{0,7056} = 1,48239$$

$$x_i = \frac{y_i - \sum_{k=i+1}^n t_{ki} x_k}{t_{ii}} \quad (i < n)$$

$$x_3 = \frac{y_3 - \sum_{k=4}^4 t_{k3} x_k}{t_{33}} = \frac{y_3 - t_{43} x_4}{t_{33}} = \frac{0,593358 - (-0,18533 * 1,48239)}{0,835376} = 1,03917$$

$$x_2 = \frac{y_2 - \sum_{k=3}^4 t_{k2} x_k}{t_{22}} = \frac{y_2 - (t_{32} x_3 + t_{42} x_4)}{t_{22}} = \frac{0,41211 - (0,102697 * 1,03917 + 0,179389 * 1,48239)}{0,907524} = 0,0434873$$

$$x_1 = \frac{y_1 - \sum_{k=2}^4 t_{k1} x_k}{t_{11}} = \frac{y_1 - (t_{21} x_2 + t_{31} x_3 + t_{41} x_4)}{t_{11}} = \frac{0,3 - (0,42 * 0,0434873 + 0,54 * 1,03917 + 0,66 * 1,48239)}{1} = -1,25779$$

$$\begin{cases} x_1 = -1,25779 \\ x_2 = 0,0434873 \\ x_3 = 1,03917 \\ x_4 = 1,48239 \end{cases}$$

Робимо перевірку:

$$\begin{cases} x_1 + 0,42x_2 + 0,54x_3 + 0,66x_4 = 0,3 \\ 0,42x_1 + x_2 + 0,32x_3 + 0,44x_4 = 0,5 \\ 0,54x_1 + 0,32x_2 + x_3 + 0,22x_4 = 0,7 \\ 0,66x_1 + 0,44x_2 + 0,22x_3 + x_4 = 0,9 \end{cases}$$

Замість іксів підставляємо корені, які знайшли

$$\begin{cases} -1,25779 + 0,42 * 0,0434873 + 0,54 * 1,03917 + 0,66 * 1,48239 = 0,300003866 \\ 0,42 * (-1,25779) + 0,0434873 + 0,32 * 1,03917 + 0,44 * 1,48239 = 0,5000015 \\ 0,54 * (-1,25779) + 0,32 * 0,0434873 + 1,03917 + 0,22 * 1,48239 = 0,700005136 \\ 0,66 * (-1,25779) + 0,44 * 0,0434873 + 0,22 * 1,03917 + 1,48239 = 0,900000412 \end{cases}$$

Результат роботи програми

```
Microsoft Visual Studio Debug Console
Your matrix A:
    1      0.42      0.54      0.66
    0.42      1      0.32      0.44
    0.54      0.32      1      0.22
    0.66      0.44      0.22      1

Your matrix T:
    1      0      0      0
    0.42  0.907524      0      0
    0.54  0.102697  0.835376      0
    0.66  0.179389 -0.185333  0.7056

Your transponent matrix T:
    1      0.42      0.54      0.66
    0  0.907524  0.102697  0.179389
    0      0  0.835376 -0.185333
    0      0      0      0.7056

Your multiply matrix T:
    1      0.42      0.54      0.66
    0.42      1      0.32      0.44
    0.54      0.32      1      0.22
    0.66      0.44      0.22      1

Your vector Y:
0.3  0.41211  0.593358  1.04598

Your vector B:
0.3  0.5  0.7  0.9

Your vector X:
-1.25779  0.0434873  1.03917  1.48239

Your answers:
0.3  0.5  0.7  0.9
```

Код програми

```
#include <iostream>
#include <cmath>
#include <vector>
#include <string>
#include <iomanip>
using namespace std;

int main()
{
    float matrix_A[4][4] = {{1, 0.42, 0.54, 0.66}, {0.42, 1, 0.32, 0.44}, {0.54, 0.32, 1,
0.22}, {0.66, 0.44, 0.22, 1}};
    float vector_B[4] = {0.3, 0.5, 0.7, 0.9};
    cout.width(12);
    cout << "Your matrix A:\n";
    for (int i = 0; i < sizeof(matrix_A)/16; i++) {
        for (int j = 0; j < sizeof(matrix_A[i]) / 4; j++) {
            cout.width(8);
            cout << matrix_A[i][j] << "    ";
        }
        cout << endl;
    }

    /// ////////////////////////////////// Находим матрицу T ////////////////////////////////// ///

    float matrix_T[4][4];
    matrix_T[0][0] = sqrt(matrix_A[0][0]);
    for (int i = 0; i < sizeof(matrix_T) / 16; i++) {
        for (int j = 0; j < sizeof(matrix_T[i]) / 4; j++) {
            if (i == 0) {
                matrix_T[j][i] = matrix_A[i][j] / matrix_T[0][0];
            }
            else if (i == j) {
                float sum_K = 0;
                int k = 0;
                while (k < i) {
                    sum_K += pow(matrix_T[j][k], 2);
                    k++;
                }
                matrix_T[j][i] = sqrt(matrix_A[i][j] - sum_K);
            }
            else if (j > i) {
                float sum_K = 0;
                int k = 0;
                while (k < i) {
                    sum_K += matrix_T[i][k] * matrix_T[j][k];
                    k++;
                }
                matrix_T[j][i] = (matrix_A[i][j] - sum_K) / matrix_T[i][i];
            }
            else {
                matrix_T[j][i] = 0;
            }
        }
    }

    cout << endl << endl;
    cout << "Your matrix T:\n";
    for (int i = 0; i < sizeof(matrix_T) / 16; i++) {
        for (int j = 0; j < sizeof(matrix_T[i]) / 4; j++) {
            cout.width(8);
            cout << matrix_T[i][j] << "    ";
        }
        cout << endl;
    }
}
```

```

}

/// ////////////////////////////////// Транспонируем матрицу T ////////////////////////////////// ///
float matrix_T_[4][4];

for (int i = 0; i < sizeof(matrix_T) / 16; i++) {
    for (int j = 0; j < sizeof(matrix_T[i]) / 4; j++) {
        matrix_T_[j][i] = matrix_T[i][j];
    }
}
cout << endl << endl;
cout.width(12);
cout << "Your transponent matrix T:\n";
for (int i = 0; i < sizeof(matrix_T_) / 16; i++) {
    for (int j = 0; j < sizeof(matrix_T_[i]) / 4; j++) {
        cout.width(8);
        cout << matrix_T_[i][j] << "    ";
    }
    cout << endl;
}

/// ////////////////////////////////// Уножаем матрицы для проверки ////////////////////////////////// ///
float matrix_Multiply[4][4];
for (int i = 0; i < sizeof(matrix_T) / 16; i++) {
    for (int j = 0; j < sizeof(matrix_T[i]) / 4; j++) {
        float sum_K = 0;
        for (int k = 0; k < 4; k++) {
            sum_K += matrix_T[i][k] * matrix_T_[k][j];
        }
        matrix_Multiply[i][j] = sum_K;
    }
}
cout << endl << endl;
cout.width(12);
cout << "Your multiply matrix T:\n";
for (int i = 0; i < sizeof(matrix_Multiply) / 16; i++) {
    for (int j = 0; j < sizeof(matrix_Multiply[i]) / 4; j++) {
        cout.width(8);
        cout << matrix_Multiply[i][j] << "    ";
    }
    cout << endl;
}

/// ////////////////////////////////// находим вектор y ////////////////////////////////// ///

float vector_Y[4];
vector_Y[0] = vector_B[0] / matrix_T[0][0];
for (int i = 1; i < sizeof(vector_B) / 4; i++) {
    int k = 0;
    float sum_K = 0;
    while (k < i) {
        sum_K += matrix_T[i][k] * vector_Y[k];
        k++;
    }
    vector_Y[i] = (vector_B[i] - sum_K) / matrix_T[i][i];
}
cout << endl << endl;
cout.width(12);
cout << "Your vector Y:\n";
for (int i = 0; i < sizeof(vector_Y) / 4; i++) {
    cout << vector_Y[i] << "    ";
}
}

```



```

/// //////////////////////////////////// находим неизвестные x ////////////////////////////////////
cout << endl << endl;
cout << "Your vector B:\n";
for (int i = 0; i < sizeof(vector_B) / 4; i++) {
    cout << vector_B[i] << " ";
}

float vector_X[4];
vector_X[3] = vector_Y[3] / matrix_T[3][3];
for (int i = sizeof(vector_Y) / 4 - 2; i >= 0 ; i--) {
    int k = i+1;
    float sum_K = 0;
    while (k < 4) {
        sum_K += matrix_T[k][i] * vector_X[k];
        k++;
    }
    vector_X[i] = (vector_Y[i] - sum_K) / matrix_T[i][i];
}
cout << endl << endl;
cout << "Your vector X:\n";
for (int i = 0; i < sizeof(vector_X) / 4; i++) {
    cout << vector_X[i] << " ";
}

/// //////////////////////////////////// Делаем проверку ////////////////////////////////////
float answers[4];
for (int i = 0; i < sizeof(matrix_T) / 16; i++) {
    float sum_K = 0;
    for (int j = 0; j < sizeof(matrix_T[i]) / 4; j++) {
        sum_K += vector_X[j] * matrix_A[i][j];
    }
    answers[i] = sum_K;
}
cout << endl << endl;
cout << "Your answers:\n";
for (int i = 0; i < sizeof(answers) / 4; i++) {
    cout << answers[i] << " ";
}
cout << endl;
return 0;
}

```