

Лабораторная работа

Метод Выбора

Выполнила: Шеховцова Е.

Постановка задачи:

Разработать программу, решающую систему линейных уравнений методом выбора.

$$A = \begin{pmatrix} 5 & 7 & 6 & 5 \\ 7 & 10 & 8 & 7 \\ 6 & 8 & 10 & 9 \\ 5 & 7 & 9 & 10 \end{pmatrix} \quad B = \begin{pmatrix} 23 \\ 32 \\ 33 \\ 31 \end{pmatrix}$$

В данной работе рассматривается метод выбора главных элементов по столбцам и используется метод Гаусса-Жордана.

Математическая модель:

$$\widetilde{a}_{ki} = \frac{a_{ki}}{a_{ii}}$$

$$\widetilde{a}_{kj} = a_{kj} - \widetilde{a}_{ki}a_{ij}$$

$$i = 1 \div (n-1),$$

$$k = 1 \div n,$$

$$j = i \div (n+1)$$

$$x_i = (a_{i(n+1)} - \sum_{j=i+1}^n a_{ij}x_j) / a_{ii}$$

Программная реализация:

```
#include <iostream>
```

```
#include <math.h>
```

```
using namespace std;
```

```
int main()
```

```
{
```

```
    float a[4][5] = {{5.0, 7.0, 6.0, 5.0, 23.0}, {7.0, 10.0, 8.0, 7.0, 32.0}, {6.0, 8.0, 10.0, 9.0, 33.0}, {5.0, 7.0, 9.0, 10.0, 31.0}};
```

```
/******
```

Прямой ход

```
*****/
```

```
for(int i = 0; i < 4; i++) {
```

```
    for(int j = 0; j < 5; j++) {
```

```
        cout << a[i][j] << "\t";
```

```
    }
```

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

```
    printf("\n");
```

```
float maximum, temp;
```

```
int index;
```

```
for(int j=0; j< 5; j++) {
```

```
    maximum=a[j][j];
```

```
    index=j;
```

```
    for(int i = j; i < 4; i++) {
```

```
        if(a[i][j]>maximum) {
```

```
            maximum=a[i][j];
```

```
            index=i;
```

```
        }
```

```
    }
```

```
for(int j1=0; j1 < 5; j1++) {
```

```
    temp = a[index][j1];
```

```
    a[index][j1] = a[j][j1];
```

```
    a[j][j1] = temp;
```

```
}
```

```
for(int k = j+1; k < 4; k++){
```

```
    float koef = a[k][j]/a[j][j];
```

```
    for(int jet = 0; jet <5; jet++) {
```

```
        if(k!=j){
```

```

        a[k][jet]-=a[j][jet]*koef;

        a[k][jet] = round(a[k][jet]*1000)/1000;

    }

}

}

}

```

```

for(int i = 0; i < 4; i++) {

    for(int j = 0; j < 5; j++) {

        cout << a[i][j] << "\t";

    }

    printf("\n");}

```

```
float x[4];
```

```
/******
```

Обратный ход

```
*****/
```

```
x[3] = round(a[3][4]/a[3][3]);
```

```
for(int i = 2; i >= 0; i--) {
```

```
    float s = 0;
```

```
    for(int j = i+1; j < 4; j++) {
```

```
        s += a[i][j]*x[j];
```

```
    }
```

```
    x[i] = (a[i][4] - s)/a[i][i];
```

```
    x[i] = round(x[i]);
```

```
}
```

```
printf("\n");
```

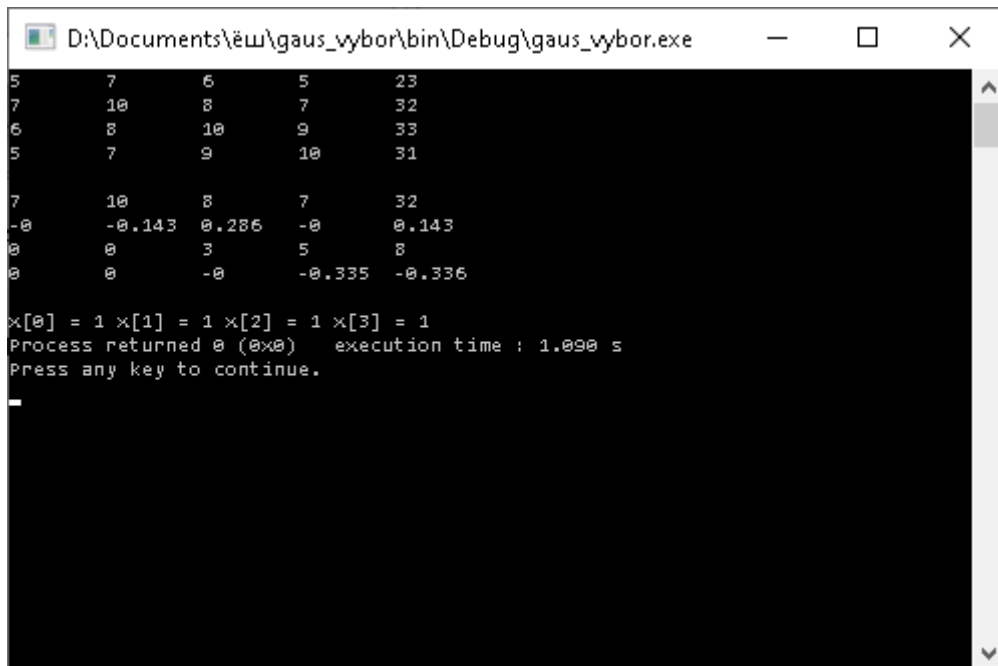
```
for(int i = 0; i < 4; i++) {
```

```
    cout << "x[" << i << "] = " << x[i] << " ";
```

```
}
```

```
return 0;  
  
}
```

Результат работы:



```
D:\Documents\ёш\gaus_vybor\bin\Debug\gaus_vybor.exe  
5      7      6      5      23  
7      10     8      7      32  
6      8      10     9      33  
5      7      9      10     31  
  
7      10     8      7      32  
-0     -0.143 0.286 -0     0.143  
0      0      3      5      8  
0      0      -0     -0.335 -0.336  
  
x[0] = 1 x[1] = 1 x[2] = 1 x[3] = 1  
Process returned 0 (0x0) execution time : 1.090 s  
Press any key to continue.  
_
```

Вывод:

При использовании данного метода необходимо было использовать округление, для этой системы уравнений, решение выбором главного элемента в столбце с использованием метода Гаусса-Жордана не является достаточно точным.