

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
НАВЧАЛЬНО-НАУКОВИЙ КОМПЛЕКС  
“ІНСТИТУТ ПРИКЛАДНОГО СИСТЕМНОГО АНАЛІЗУ”  
НАЦІОНАЛЬНОГО ТЕХНІЧНОГО УНІВЕРСИТЕТУ УКРАЇНИ  
“КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ІМЕНІ ІГОРЯ  
СІКОРСЬКОГО”  
КАФЕДРА МАТЕМАТИЧНИХ МЕТОДІВ СИСТЕМНОГО АНАЛІЗУ

Лабораторна робота №3  
**ІТЕРАЦІЙНІ МЕТОДИ РОЗ’ЯЗАННЯ СИСТЕМ  
ЛІНІЙНИХ АЛГЕБРАЇЧНИХ РІВНЯНЬ**  
**Варіант-27**

Виконав:  
Терещенко Денис, КА-96  
Прийняла:  
Шубенкова І. А.

КИЇВ - 2020

## ЗВІТ

**Мета роботи:** засвоїти ітераційні методи, які приходять на допомогу, якщо прямі методи застосувати неможливо.

**Хід роботи:** застосувати студентам для непарних номерів за списком групи – метод простої ітерації, для парних – метод Зейделя.

### 1. Постановка завдання.

1. Запрограмувати один з ітераційних методів, що зводяться до формули:

$$x_{k+1} = Bx_k + d$$

2. Для методу простої ітерації достатньою умовою збігу ітерацій є наявність діагональної переваги у системі. Вручну спробуйте перевести задану систему рівнянь до виду "з діагональною перевагою".

### 2. Аналітична частина.

Завдання варіанту:

$$A = \begin{bmatrix} 7.03 & 0.94 & 1.13 & 1.135 & -0.81 \\ 1.26 & 3.39 & 1.3 & -1.63 & -1.53 \\ 0.81 & -2.46 & 6.21 & 2.1 & -1.067 \\ 1.345 & 0.16 & 2.1 & 5.33 & 16 \\ 1.29 & 0.87 & 1.333 & -8 & 15 \end{bmatrix} \quad b = \begin{bmatrix} 2.1 \\ 0.84 \\ -3.44 \\ -0.92 \\ -1.47 \end{bmatrix}$$

Реалізуємо розв'язок заданої системи за методом простих ітерацій. Критерієм закінчення ітераційного процесу є норма вектору нев'язки  $\|\xi_{k+1}\| \leq \varepsilon = 10^{-5}$ . Підрахувавши, отримали, що  $\alpha = 38.501 > 1$ . Тому м-цю вигідно привести до системи з діагональною перевагою. Тобто:

$$A^* = \begin{bmatrix} 7.03 & 0.94 & 1.13 & 1.135 & -0.81 \\ 1.26 & 3.39 & 1.3 & -1.637 & -1.53 \\ 0.81 & -2.46 & 6.21 & 2.1 & -1.067 \\ 1.345 & 0.16 & 2.1 & 5.33 & 16 \\ 2.635 & 1.03 & 3.433 & -2.67 & 31 \end{bmatrix} \quad b^* = \begin{bmatrix} 2.1 \\ 0.84 \\ -3.44 \\ -0.92 \\ -2.39 \end{bmatrix}$$

### 3. Лістинг програми.

```
1  #include <iostream>
   #include <iomanip>
3  #include <cmath>
   #define e 0.00001
5  using namespace std;
   class Matr
7  {
   private:
9      int size;
      double **mas;
11     double *mas1;
   public:
13     Matr()
        {
15         size = 0;
         mas = NULL;
17         mas1 = NULL;
        }
19     Matr(int l)
        {
21         size = l;
         mas = new double*[l];
23         for (int i = 0; i < l; i++)
             mas[i] = new double[l];
25         mas1 = new double[l];
        }
27     void Add()
        {
29         for (int i = 0; i < size; i++)
             for (int j = 0; j < size; j++)
31                 cin >> mas[i][j];
         for (int i = 0; i < size; i++)
33         {
             cin >> mas1[i];
35         }
        }
37     }
   void Print()
39   {
       for (int i = 0; i < size; i++)
41       {
           for (int j = 0; j < size; j++)
43           {
               cout << setw(4) << mas[i][j] << " ";
45           }
           cout << " " << mas1[i] << endl;
47       }
   }
```

```

49 void Preob()
50 {
51     double temp = 0;
52     for (int k = 0; k < size; k++)
53     {
54         for (int i = 0; i < size; i++)
55         {
56             temp = mas[i][i]*(-1);
57             mas1[i] /= temp;
58             for (int j = 0; j <= size; j++)
59             {
60                 mas[i][j] /= temp;
61             }
62         }
63     }
64     for (int i = 0; i < size; i++)
65     {
66         mas1[i] *= -1;
67         for (int j = 0; j < size; j++)
68             mas[i][i] = 0;
69     }
70 }
71 double Norma(double **mas)
72 {
73     double sum = 0, max = 0;
74     for (int i = 0; i < size; i++)
75     {
76         for (int j = 0; j < size; j++)
77         {
78             sum += fabs(mas[i][j]);
79             if (sum > max) max = sum;
80         }
81         sum = 0;
82     }
83     return max;
84 }
85 double Pogr()
86 {
87     double eps = 0;
88     double norm = Norma(mas);
89     eps = -((1 - norm) / norm)*e;
90     cout<< eps;
91     return eps;
92 }
93 void Itera()
94 {
95     double *x = new double[size];
96     double *x0 = new double[size];
97     double *E = new double[size];
98     double max = 0, per = 0;

```

```

99     per = Pogr();
100     for (int i = 0; i < size; i++)
101         x0[i] = mas1[i];
102     int counter = 0;
103     do
104     {
105         for (int i = 0; i < size; i++)
106         {
107             x[i] = 0;
108             for (int j = 0; j < size; j++)
109             {
110                 x[i] += mas[i][j] * x0[j];
111             }
112             x[i] += mas1[i];
113             E[i] = fabs(x[i] - x0[i]);
114         }
115         max = 0;
116         for (int i = 0; i < size; i++)
117         {
118             if (max < E[i]) max = E[i];
119             x0[i] = x[i];
120         }
121         cout<< endl <<" Iter:" << counter ;
122         cout<< endl<< "x:";
123             for (int i = 0; i < size; i++) cout<< " "<< x[i];
124         cout<< endl<< "vec_xi:";
125             for (int i = 0; i < size; i++) cout<< " "<< E[i];
126
127         cout<<endl << "abs_xi = " << max;
128         counter++;
129     } while (max > per && counter < 50);
130     cout << endl << "Count of iterations: " << counter << endl;
131     for (int i = 0; i < size; i++)
132         cout << "x" << i + 1 << "=" << x[i] << " " << endl;
133     delete[] x;
134     delete[] x0;
135     delete[] E;
136 }
137 };
138
139 int main()
140 {
141     setlocale(LC_ALL, "rus");
142     int n;
143     cout << "Insert number of variables: ";
144     cin >> n;
145     Matr a(n);
146     cout << "Insert matrix A:" << endl;
147     a.Add();
148     a.Preob();

```

```

149     a.Itera();
        cout << endl;
151     system("pause");
        return 0;
153 }

```

## 4. Результати роботи.

Insert number of variables: 5

Insert matrix A:

7.03 0.94 1.13 1.135 -0.81

1.26 3.39 1.3 -1.63 -1.53

0.81 -2.46 6.21 2.1 -1.067

1.345 0.16 2.1 5.33 16

2.635 1.03 3.433 -2.67 31

2.1

0.84

-3.44

-0.92

-2.39

7.28131e-06

Iter:0

x: 0.373613 0.231396 -0.449628 0.194261 -0.0642425

vec\_xi: 0.0748932 0.0163914 0.104317 0.366869 0.0128543

abs\_xi = 0.366869

Iter:1

x: 0.301287 0.345758 -0.587743 0.0961662 -0.050018

vec\_xi: 0.0723263 0.114361 0.138115 0.0980944 0.0142245

abs\_xi = 0.138115

Iter:2

x: 0.325672 0.384858 -0.497391 0.122701 -0.0408237

vec\_xi: 0.0243854 0.0391004 0.0903525 0.026535 0.00919435

abs\_xi = 0.0903525

Iter:3

x: 0.302696 0.358054 -0.492476 0.0521751 -0.051916

vec\_xi: 0.0229762 0.0268037 0.00491492 0.0705261 0.0110923

abs\_xi = 0.0705261

Iter:4

x: 0.315598 0.325792 -0.478153 0.0901388 -0.055691

vec\_xi: 0.0129024 0.032262 0.0143226 0.0379637 0.00377509

abs\_xi = 0.0379637

Iter:5

x: 0.311046 0.332054 -0.506103 0.0935408 -0.0540321

```

vec_xi: 0.00455261 0.00626214 0.0279497 0.00340191 0.0016589
abs_xi = 0.0279497
Iter:6
x: 0.314343 0.346849 -0.503894 0.100534 -0.050465
vec_xi: 0.00329719 0.0147947 0.0022091 0.0069931 0.00356711
abs_xi = 0.0147947
Iter:7
x: 0.311292 0.349749 -0.500215 0.0876793 -0.0508792
vec_xi: 0.00305137 0.00289975 0.00367872 0.0128545 0.000414158
abs_xi = 0.0128545
Iter:8
x: 0.31234 0.343105 -0.494393 0.0881561 -0.0522307
vec_xi: 0.00104861 0.0066443 0.00582248 0.0004768 0.00135152
abs_xi = 0.0066443
Iter:9
x: 0.31206 0.340101 -0.497555 0.089854 -0.0527028
vec_xi: 0.000280179 0.00300328 0.00316227 0.0016979 0.000472096
abs_xi = 0.00316227
Iter:10
x: 0.312641 0.342021 -0.499363 0.092678 -0.0520828
vec_xi: 0.000581357 0.00192013 0.00180844 0.00282395 0.000620036
abs_xi = 0.00282395
Iter:11
x: 0.312291 0.344137 -0.499527 0.0913249 -0.0517525
vec_xi: 0.000350546 0.00211509 0.000163624 0.00135309 0.000330282
abs_xi = 0.00211509
Iter:12
x: 0.312291 0.343828 -0.498129 0.0904228 -0.0518914
vec_xi: 4.69963e-10 0.000308499 0.0013979 0.000902031 0.0001389
abs_xi = 0.0013979
Iter:13
x: 0.312237 0.342796 -0.49797 0.0902983 -0.0521136
vec_xi: 5.38186e-05 0.00103248 0.000158962 0.000124548 0.000222247
abs_xi = 0.00103248
Iter:14
x: 0.312344 0.342594 -0.498368 0.0909474 -0.0521031
vec_xi: 0.000107005 0.000201147 0.00039805 0.000649104 1.05486e-05
abs_xi = 0.000649104
Iter:15
x: 0.312331 0.343024 -0.49868 0.0910516 -0.0520055
vec_xi: 1.27045e-05 0.00042974 0.00031133 0.000104201 9.75754e-05
abs_xi = 0.00042974

```

```

Iter:16
x: 0.312318 0.343242 -0.498526 0.0908716 -0.0519753
vec_xi: 1.29992e-05 0.000218252 0.000153421 0.000179941 3.02534e-05
abs_xi = 0.000218252
Iter:17
x: 0.312297 0.343116 -0.498372 0.0907171 -0.0520139
vec_xi: 2.13064e-05 0.000126868 0.0001542 0.000154536 3.86349e-05
abs_xi = 0.000154536
Iter:18
x: 0.31231 0.342973 -0.498374 0.0907815 -0.0520383
vec_xi: 1.26761e-05 0.000142955 1.85775e-06 6.44077e-05 2.43601e-05
abs_xi = 0.000142955
Iter:19
x: 0.312316 0.342989 -0.498458 0.0908565 -0.0520288
vec_xi: 6.20809e-06 1.59754e-05 8.4249e-05 7.49506e-05 9.42544e-06
abs_xi = 8.4249e-05
Iter:20
x: 0.312316 0.343059 -0.498476 0.0908593 -0.0520141
vec_xi: 3.91208e-07 7.02926e-05 1.82074e-05 2.85364e-06 1.47268e-05
abs_xi = 7.02926e-05
Iter:21
x: 0.312311 0.343074 -0.498447 0.0908201 -0.0520142
vec_xi: 5.23624e-06 1.48555e-05 2.93597e-05 3.92433e-05 1.06672e-07
abs_xi = 3.92433e-05
Iter:22
x: 0.312311 0.343045 -0.498427 0.0908097 -0.0520209
vec_xi: 3.82068e-07 2.823e-05 1.98201e-05 1.0372e-05 6.67984e-06
abs_xi = 2.823e-05
Iter:23
x: 0.312312 0.34303 -0.498436 0.0908229 -0.052023
vec_xi: 1.49374e-06 1.54606e-05 8.77335e-06 1.31868e-05 2.11781e-06
abs_xi = 1.54606e-05
Iter:24
x: 0.312313 0.343038 -0.498447 0.0908328 -0.0520205
vec_xi: 1.10445e-06 8.19397e-06 1.11425e-05 9.90124e-06 2.49407e-06
abs_xi = 1.11425e-05
Iter:25
x: 0.312313 0.343048 -0.498447 0.0908292 -0.0520188
vec_xi: 6.1579e-07 9.74885e-06 1.82146e-07 3.62146e-06 1.7206e-06
abs_xi = 9.74885e-06
Iter:26
x: 0.312312 0.343047 -0.498441 0.0908238 -0.0520194

```



```
vec_xi: 5.49887e-07 8.0571e-07 5.46246e-06 5.37404e-06 6.03655e-07  
abs_xi = 5.46246e-06  
Count of iterations: 27  
x1=0.312312  
x2=0.343047  
x3=-0.498441  
x4=0.0908238  
x5=-0.0520194
```

**Висновок.** Остаточно отримали за 27 ітерацій відповідь, що збігається з точним роз'язком системи, отриманим раніше.

$$x_1 = 0.312312 \quad x_2 = 0.343047 \quad x_3 = -0.498441 \quad x_4 = 0.0908238 \quad x_5 = -0.0520194$$

Тож, реалізацію методу простих ітерацій можна вважжати коректною.