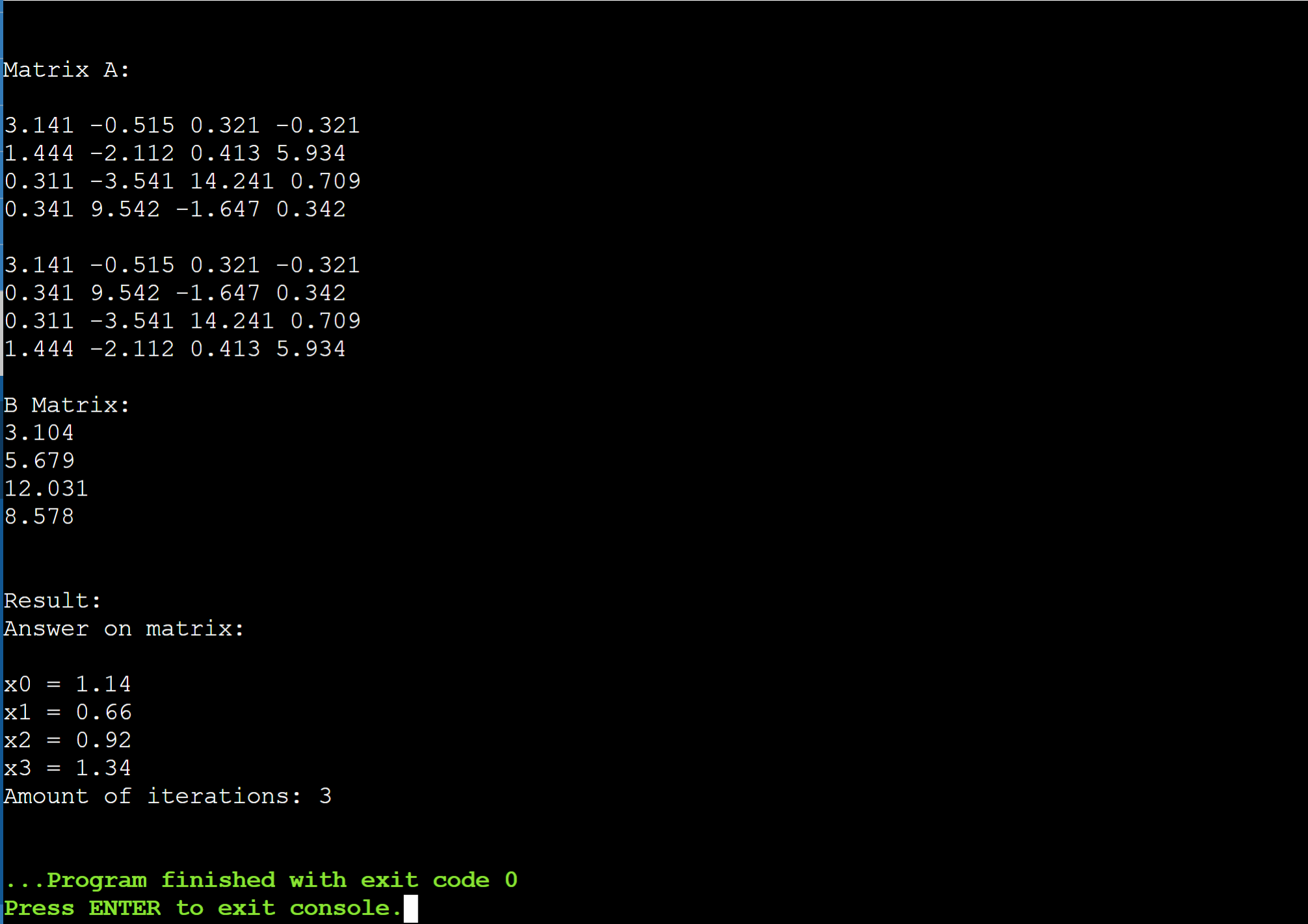
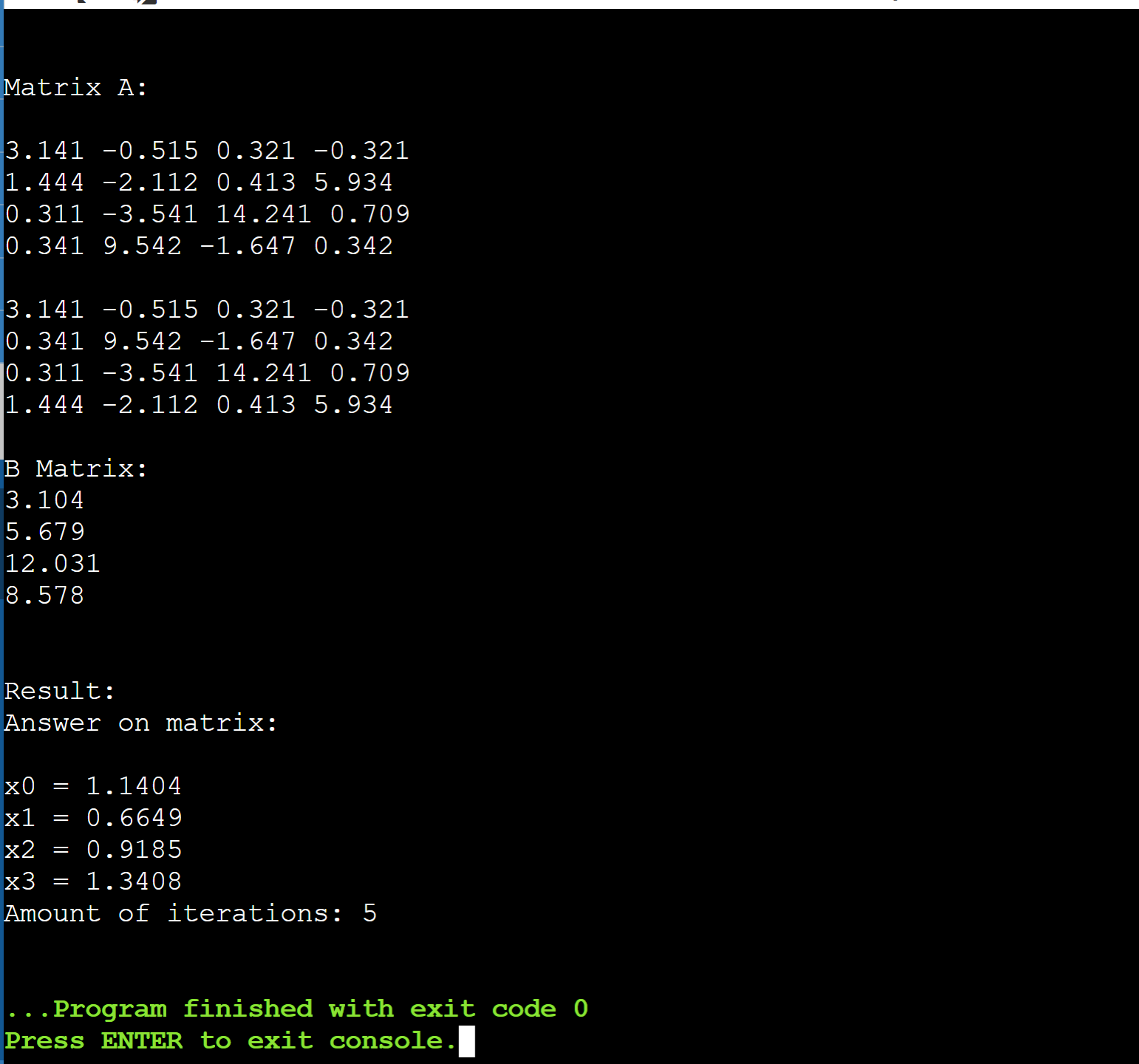


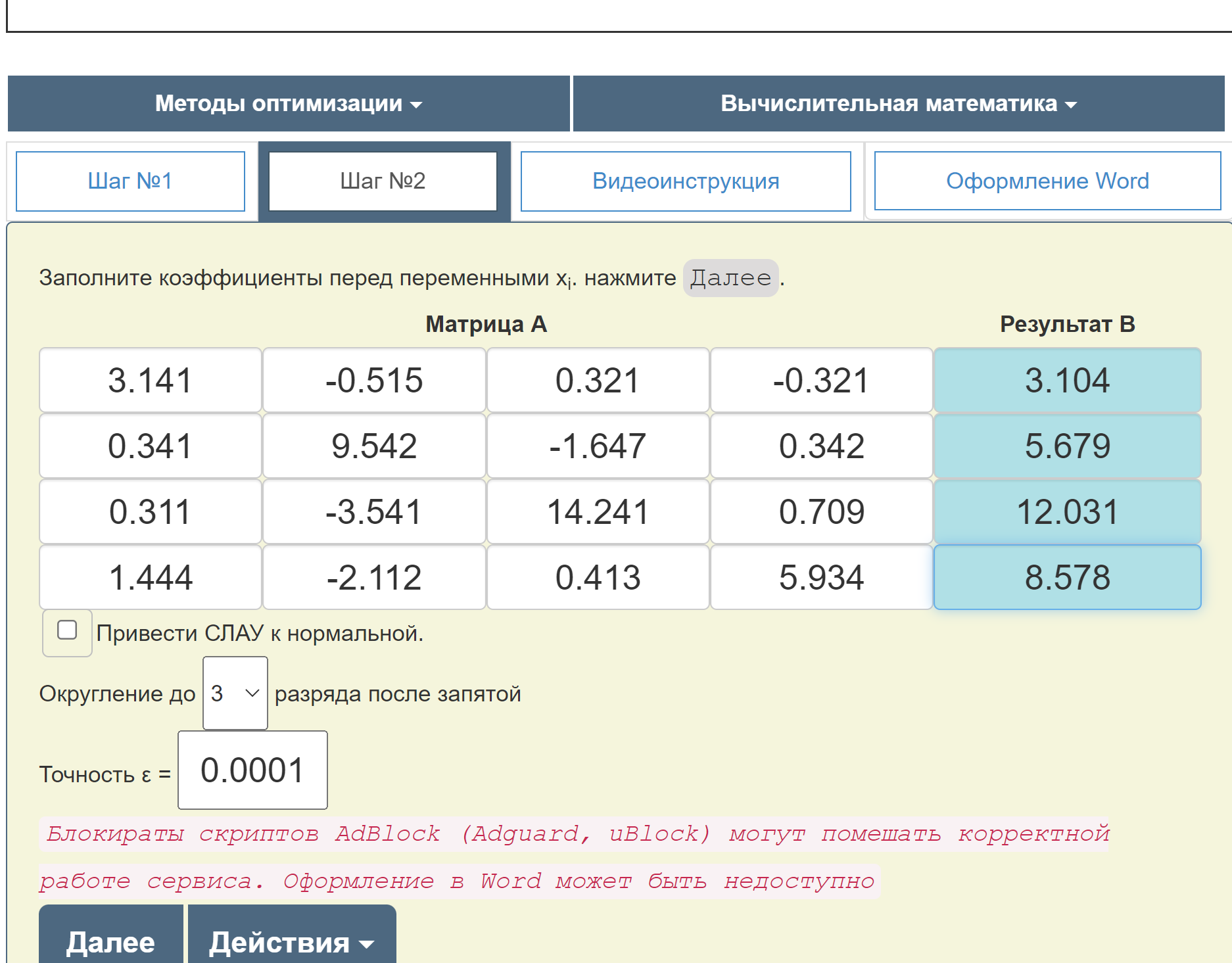
**Код завдання:**

#include <iostream>  
#include <cmath>  
  
using namespace std;  
  
// Умова закінчення  
bool lowerThanEpsilon(double xk[10], double xkp[10], int n, double eps)  
{  
 double norm = 0;  
 for (int i = 0; i < n; i++)  
 norm += (xk[i] - xkp[i]) \* (xk[i] - xkp[i]);  
 return (sqrt(norm) < eps);  
}  
  
double zeydelMethod(double x, double eps)  
{  
 int i = 0;  
 double neweps = eps;  
 while (neweps < 1)  
 {  
 i++;  
 neweps \*= 10;  
 }  
 int okr = pow(double(10), i);  
 x = int(x \* okr + 0.5) / double(okr);  
  
 return x;  
}  
  
bool highestOnDiagonal(double a[4][4])  
{  
 int i, j, k = 1;  
 double sum;  
 for (i = 0; i < 4; i++) {  
 sum = 0;  
 for (j = 0; j < 4; j++) sum += abs(a[i][j]);  
 sum -= abs(a[i][i]);  
 if (sum > a[i][i])  
 {  
 k = 0;  
 cout << a[i][i] << " < " << sum << endl;  
 }  
 else  
 {  
 cout << a[i][i] << " > " << sum << endl;  
 }  
  
 }  
  
 return (k == 1);  
  
}  
  
int main()  
{  
  
 double eps = 0.1, x[4], p[4];  
 double a[4][4] = {  
 {3.141, -0.515, 0.321, -0.321},  
 {1.444, -2.112, 0.413, 5.934},  
 {0.311, -3.541, 14.241, 0.709},  
 {0.341, 9.542, -1.647, 0.342 }  
 };  
 double b[4] = {3.104, 5.679, 12.031, 8.578};  
 int i, j, m = 0;  
 int method;  
   
 cout << endl << endl;  
 cout << "Matrix A: " << endl << endl;  
 for (i = 0; i < 4; i++)  
 {  
 for (j = 0; j < 4; j++)  
 cout << a[i][j] << " ";  
 cout << endl;  
 }  
 swap(a[1], a[3]);  
 cout << endl;  
 for (i = 0; i < 4; i++)  
 {  
 for (j = 0; j < 4; j++)  
 cout << a[i][j] << " ";  
 cout << endl;  
 }  
 cout << endl;  
  
 cout << "B Matrix: " << endl;  
 for (i = 0; i < 4; i++)  
 {  
 cout << b[i] << " ";  
 cout << endl;  
 }  
 cout << endl << endl;  
  
  
 /\*  
 Хід методу, де:  
 a[n][n] - Матриця коефіцієнтів  
 x[n], p[n] - Теперешнє и попереднє рішення  
 b[n] - Стовпчик правих чисел  
 \*/  
  
 for (int i = 0; i < 4; i++)  
 x[i] = 1;  
  
 cout << "Result: " << endl;  
 do  
 {  
 for (int i = 0; i < 4; i++)  
 p[i] = x[i];  
 for (int i = 0; i < 4; i++)  
 {  
 double var = 0;  
 for (int j = 0; j < 4; j++)  
 if (j != i) var += (a[i][j] \* x[j]);  
  
 x[i] = (b[i] - var) / a[i][i];  
 }  
 m++;  
 } while (!lowerThanEpsilon(x, p, 4, eps));  
  
  
  
 cout << "Answer on matrix:" << endl << endl;  
 for (i = 0; i < 4; i++) cout << "x" << i << " = " << zeydelMethod(x[i], eps) << "" << endl;  
 cout << "Amount of iterations: " << m << endl;  
  
 return 0;  
}

**Результат:**









**Висновок:**

У процесі роботи над завданням я використав мову програмування С++. Беручи до уваги суть завдання, використав метод Зейделя, що представляє собою модифікацію метода ітерацій. Відповідно до завдання, спочатку трансформував матрицю порівняв кількість ітерацій в залежності від вказаної величини eps. При eps = 0.01 отримав кількість ітерацій = 3, при eps = 0.0001, кількість ітерацій – 5. Звірив свої результати з результатами отриманими від онлайн калькулятора і вони збіглись. Результатом задоволений.