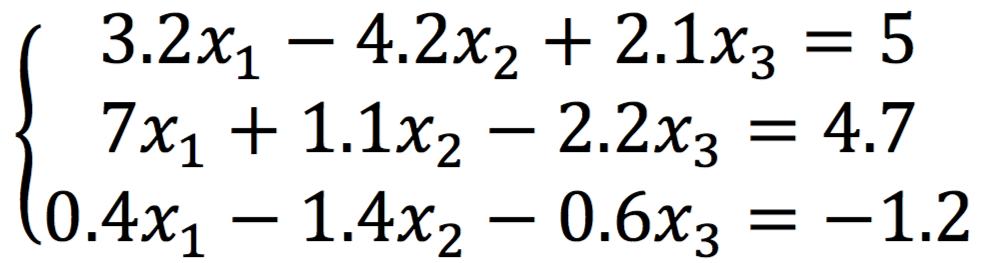
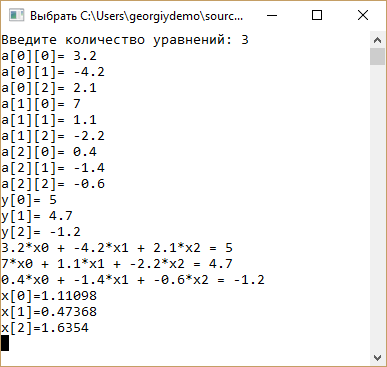
Условие:

Выполнить программную реализацию метода Гаусса для решения системы линейных уравнений с точностью до третьего знака после запятой:

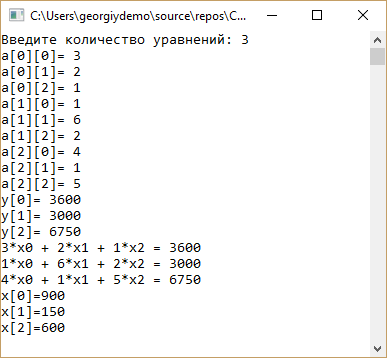


Исходный код программы:  
  
#include **"stdafx.h"**#include **<iostream>  
using namespace** std;  
*// Вывод системы уравнений***void** sysout(**double** \*\*a, **double** \*y, **int** n)  
{  
 **for** (**int** i = 0; i < n; i++)  
 {  
 **for** (**int** j = 0; j < n; j++)  
 {  
 cout << a[i][j] << **"\*x"** << j;  
 **if** (j < n - 1)  
 cout << **" + "**;  
 }  
 cout << **" = "** << y[i] << endl;  
 }  
 **return**;  
}  
**double** \* gauss(**double** \*\*a, **double** \*y, **int** n)  
{  
 **double** \*x, max;  
 **int** k, index;  
 **const double** eps = 0.00001; *// точность* x = **new double**[n];  
 k = 0;  
 **while** (k < n)  
 {  
 *// Поиск строки с максимальным a[i][k]* max = abs(a[k][k]);  
 index = k;  
 **for** (**int** i = k + 1; i < n; i++)  
 {  
 **if** (abs(a[i][k]) > max)  
 {  
 max = abs(a[i][k]);  
 index = i;  
 }  
 }  
 *// Перестановка строк* **if** (max < eps)  
 {  
 *// нет ненулевых диагональных элементов* cout << **"Решение получить невозможно из-за нулевого столбца "**;  
 cout << index << **" матрицы A"** << endl;  
 **return** 0;  
 }  
 **for** (**int** j = 0; j < n; j++)  
 {  
 **double** temp = a[k][j];  
 a[k][j] = a[index][j];  
 a[index][j] = temp;  
 }  
 **double** temp = y[k];  
 y[k] = y[index];  
 y[index] = temp;  
 *// Нормализация уравнений* **for** (**int** i = k; i < n; i++)  
 {  
 **double** temp = a[i][k];  
 **if** (abs(temp) < eps) **continue**; *// для нулевого коэффициента пропустить* **for** (**int** j = 0; j < n; j++)  
 a[i][j] = a[i][j] / temp;  
 y[i] = y[i] / temp;  
 **if** (i == k) **continue**; *// уравнение не вычитать само из себя* **for** (**int** j = 0; j < n; j++)  
 a[i][j] = a[i][j] - a[k][j];  
 y[i] = y[i] - y[k];  
 }  
 k++;  
 }  
 *// обратная подстановка* **for** (k = n - 1; k >= 0; k--)  
 {  
 x[k] = y[k];  
 **for** (**int** i = 0; i < k; i++)  
 y[i] = y[i] - a[i][k] \* x[k];  
 }  
 **return** x;  
}  
**int** main()  
{  
 **double** \*\*a, \*y, \*x;  
 **int** n;  
 system(**"chcp 1251"**);  
 system(**"cls"**);  
 cout << **"Введите количество уравнений: "**;  
 cin >> n;  
 a = **new double**\*[n];  
 y = **new double**[n];  
 **for** (**int** i = 0; i < n; i++)  
 {  
 a[i] = **new double**[n];  
 **for** (**int** j = 0; j < n; j++)  
 {  
 cout << **"a["** << i << **"]["** << j << **"]= "**;  
 cin >> a[i][j];  
 }  
 }  
 **for** (**int** i = 0; i < n; i++)  
 {  
 cout << **"y["** << i << **"]= "**;  
 cin >> y[i];  
 }  
 sysout(a, y, n);  
 x = gauss(a, y, n);  
 **for** (**int** i = 0; i < n; i++)  
 cout << **"x["** << i << **"]="** << x[i] << endl;  
 cin.get(); cin.get();  
 **return** 0;  
}

Скриншоты программы:



а) Решение исходной задачи



б) Пример решения с альтернативными данными