Федеральное государственное бюджетное образовательное учреждение «Сибирский государственный университет телекоммуникаций и информатики»

Кафедра ПМиК

Лабораторная работа №1

по дисциплине

«Алгоритмы и вычислительные методы оптимизации»

Выполнил: студент III курса

Группы ИП-813

Бурдуковский И.А

Проверила:

Галкина М.Ю.

Новосибирск 2021

Оглавление

[Задание на лабораторную работу 3](#_Toc40794109)

[Текст программы 4](#_Toc40794110)

[Результаты тестирования 10](#_Toc40794111)

# Задание на лабораторную работу

Написать программу, находящую решение системы линейных уравнений методом Жордана-Гаусса. Программа должна выводить промежуточные матрицы после каждого шага исключений и решение системы.

# Текст программы

**Zhordan-Gauss.cpp**

#include <fstream>

#include <conio.h>

#include <iomanip>

#include <cstdio>

#include <cstring>

#include <cstdlib>

#include <iostream>

#include <sstream>

using namespace std;

struct Fraction

{

long long numerator;

long long denominator = 1;

};

void print(Fraction \*\*arr,int n,int m)

{

for (int i=0;i<m;i++)

{

for (int j=0;j<n-1;j++)

{

if(arr[i][j].denominator==1 || arr[i][j].numerator==0){

cout << arr[i][j].numerator << " ";

}

else

cout<<arr[i][j].numerator << "/" << arr[i][j].denominator<<" ";

}

cout<<"| ";

if(arr[i][n-1].denominator==1 || arr[i][n-1].numerator==0){

cout<<arr[i][n-1].numerator<<endl;

}

else

cout<<arr[i][n-1].numerator << "/" << arr[i][n-1].denominator<<endl;

}

cout<<endl;

}

long long NOD(long long a, long long b)

{

if(b == 0) return a;

else return NOD(b, a % b);

}

long long divide(Fraction \*\*arr,int n,int m){

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

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

if(arr[i][j].denominator != 0)

if(arr[i][j].numerator % arr[i][j].denominator == 0){

arr[i][j].numerator /= arr[i][j].denominator;

arr[i][j].denominator = 1;

}

else{

long long d = NOD(arr[i][j].denominator,arr[i][j].numerator);

if(d!=0){

arr[i][j].denominator /= d;

arr[i][j].numerator /= d;

}

}

}

}

}

void Jordan\_Gauss(Fraction \*\*arr,int n,int m){

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

if(arr[i][i].numerator == 0){

continue;

}

if(arr[i][i].numerator/arr[i][i].denominator < 0){

for(int j = i; j < n; j++){

arr[i][j].numerator \*= -1;

}

}

if(arr[i][i].numerator/arr[i][i].denominator != 1 || arr[i][i].numerator % arr[i][i].denominator != 0){

long long num = arr[i][i].numerator;

long long den = arr[i][i].denominator;

for(int j = i; j < n; j++){

arr[i][j].numerator \*= den;

arr[i][j].denominator \*= num;

}

}

long long ugol\_num = arr[i][i].numerator;

long long ugol\_den = arr[i][i].denominator;

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

if(j==i){

continue;

}

arr[j][i].numerator \*= ugol\_den;

arr[j][i].denominator \*= ugol\_num;

}

divide(arr,n,m);

for(int v = 0; v < m; v++){

if(v == i){

continue;

}

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

long long num = arr[v][i].numerator \* arr[i][j].numerator;

long long den = arr[v][i].denominator \* arr[i][j].denominator;

if (den != arr[v][j].denominator){

long long buf = den;

num \*= arr[v][j].denominator;

den \*= arr[v][j].denominator;

arr[v][j].numerator \*= buf;

arr[v][j].denominator \*= buf;

}

arr[v][j].numerator -= num;

}

arr[v][i].numerator = 0;

arr[v][i].denominator = 0;

if(arr[v][n-1].numerator != 0){

int zero\_counter = 0;

for (int j = 0; j <n-1;j++){

if (arr[v][j].numerator!=0){

break;

}

zero\_counter++;

}

if(zero\_counter==n-1){

divide(arr,n,m);

cout << endl;

print(arr,n,m);

cout << "There are no resolves here!"<<endl;

return;

}

}

}

divide(arr,n,m);

cout << endl;

print(arr,n,m);

}

print(arr,n,m);

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

if(arr[i][i].numerator==0){

continue;

}

cout << "x" << i+1 << " = ";

if(arr[i][n-1].numerator==0){

cout << "0 ";

}

else{

cout << arr[i][n-1].numerator;

if (arr[i][n-1].denominator != 1){

cout << "/" << arr[i][n-1].denominator;

}

}

for(int j = i+1; j < n-1; j++){

if (arr[i][j].numerator == 0){

continue;

}

if (arr[i][j].numerator < 0){

if (arr[i][j].denominator < 0){

cout << " - " << arr[i][j].numerator \* -1 << "/" << arr[i][j].denominator\*-1;

}

else{

if (arr[i][j].denominator != 1){

cout << " + " << arr[i][j].numerator \* -1 << "/" << arr[i][j].denominator;

}

else{

cout << " + " << arr[i][j].numerator \* -1;

}

}

}

if (arr[i][j].numerator > 0){

if (arr[i][j].denominator < 0){

cout << " + " << arr[i][j].numerator << "/" << arr[i][j].denominator\*-1;

}

else{

if (arr[i][j].denominator != 1){

cout << " - " << arr[i][j].numerator << "/" << arr[i][j].denominator;

}

else{

cout << " - " << arr[i][j].numerator;

}

}

}

cout << " \* x" <<j+1;

}

cout << endl;

}

return;

}

int main()

{

int n,m;

string filename;

cout << "Enter file name:" << endl;

cin >> filename;

ifstream input(filename);

if(!input){

cout << "Invalid file name" << endl;

return 0;

}

input >> n;

input >> m;

Fraction \*\*arr=new Fraction\*[m];

for (int i=0;i<m;i++)

{

arr[i]=new Fraction[n];

}

for (int i=0;i<m;i++)

{

for (int j=0;j<n;j++)

{

input>>arr[i][j].numerator;

}

}

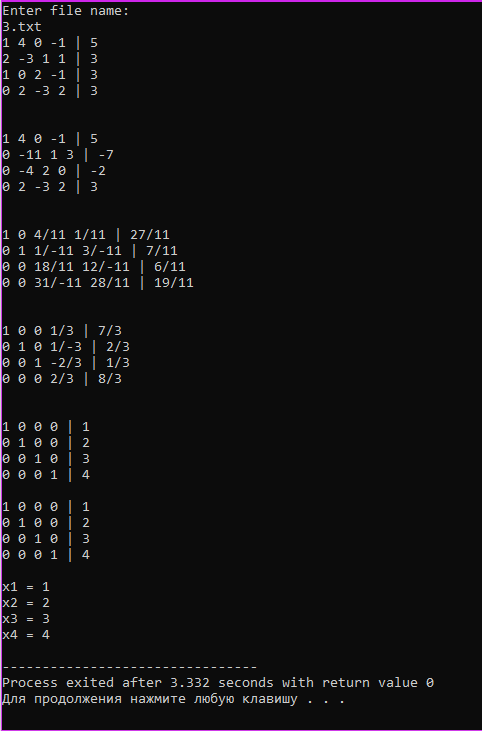
print(arr,n,m);

Jordan\_Gauss(arr,n,m);

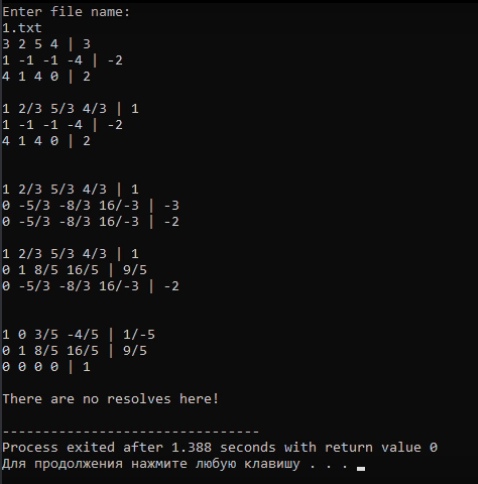
return 0;

}

# Результаты тестирования

1. Единственное решение   
   

2. Нет решений



3. Бесконечно много решений

