МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение

высшего образования

**«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»**

**(ФГБОУ ВО «КубГУ»)**

**Факультет компьютерных технологий и прикладной математики**

**Кафедра информационных технологий**

**ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 9**

**по дисциплине  
 «АЛГОРИТМЫ И АНАЛИЗ СЛОЖНОСТИ»**

Выполнил студент группы 24/1                                       А.А.Иванов

Направление подготовки  02.03.03  Математическое обеспечение и администрирование информационных систем

Курс    2

Отчет принял доктор физико-математических наук, профессор                                                                                       А.И. Миков

Краснодар

2022 г.

**Задание:** найти зависимость средней временной сложности алгоритма Iso проверки изоморфизма графов T(n), при n от 1 до 20 с шагом 1.

**Решение:** напишем несколько функций, которые будут делать следующие - создаётся граф G1 через матрицу инцедентности, в ней меняем местами строки и столбцы, чтобы получить новый граф G2, проверяем их не изоморфизм.

**Код программы:**

#include <iostream>

#include <string>

#include <algorithm>

#include <random>

#include <cstdlib>

#include <ctime>

using namespace std;

int kolop = 0;

void G1g(int\*\* m, int n)

{

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

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

{

if (i <= j)

m[i][j] = rand() % 2;

else

m[i][j] = -1;

}

}

void Dopolnenie(int\*\* m, int n)

{

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

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

if (m[i][j] == -1)

m[i][j] = m[j][i];

}

void MenMest(int\*\* m, int a1, int a2, int n)

{

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

{

int nomer = m[a1][i];

m[a1][i] = m[a2][i];

m[a2][i] = nomer;

}

}

void G2g(int\*\* m2, int n)

{

int r = rand() % (n);

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

{

int a1 = rand() % (n - 1);

int a2 = rand() % (n - 1);

if (a1 == a2)

{

if (a1 == (n - 1))

a2--;

else

a2++;

}

MenMest(m2, a1, a2, n);

}

}

bool Poisk(int\*\* m, int\*\* m2, int n)

{

int kol = 0;

kolop++;

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

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

{

int b = 0;

kolop++;

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

{

if (m[i][j] == m2[i][j])

b++;

kolop += 3;

}

if (b == n)

{

kol++;

kolop += 2;

break;

}

}

kolop++;

if (kol == n)

return true;

return false;

}

void G2Poln(int\*\* m, int\*\* m2, int n)

{

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

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

m2[i][j] = m[i][j];

}

int main()

{

setlocale(LC\_ALL, "rus");

srand(time(NULL));

for (int i = 1; i < 21; i++)

{

int n = i;

cout << "Кол-во вершин в графе - " << n << endl;

int\*\* G1 = new int\* [n];

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

G1[i] = new int[n];

int\*\* G2 = new int\* [n];

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

G2[i] = new int[n];

G1g(G1, n);

Dopolnenie(G1, n);

G2Poln(G1, G2, n);

G2g(G2, n);

bool f = Poisk(G1, G2, n);

if (f == true)

cout << "Ваши графы изоморфны" << endl;

else

cout << "Ваши графы не изоморфны" << endl;

cout << "Кол-во операций - " << kolop << endl << endl;

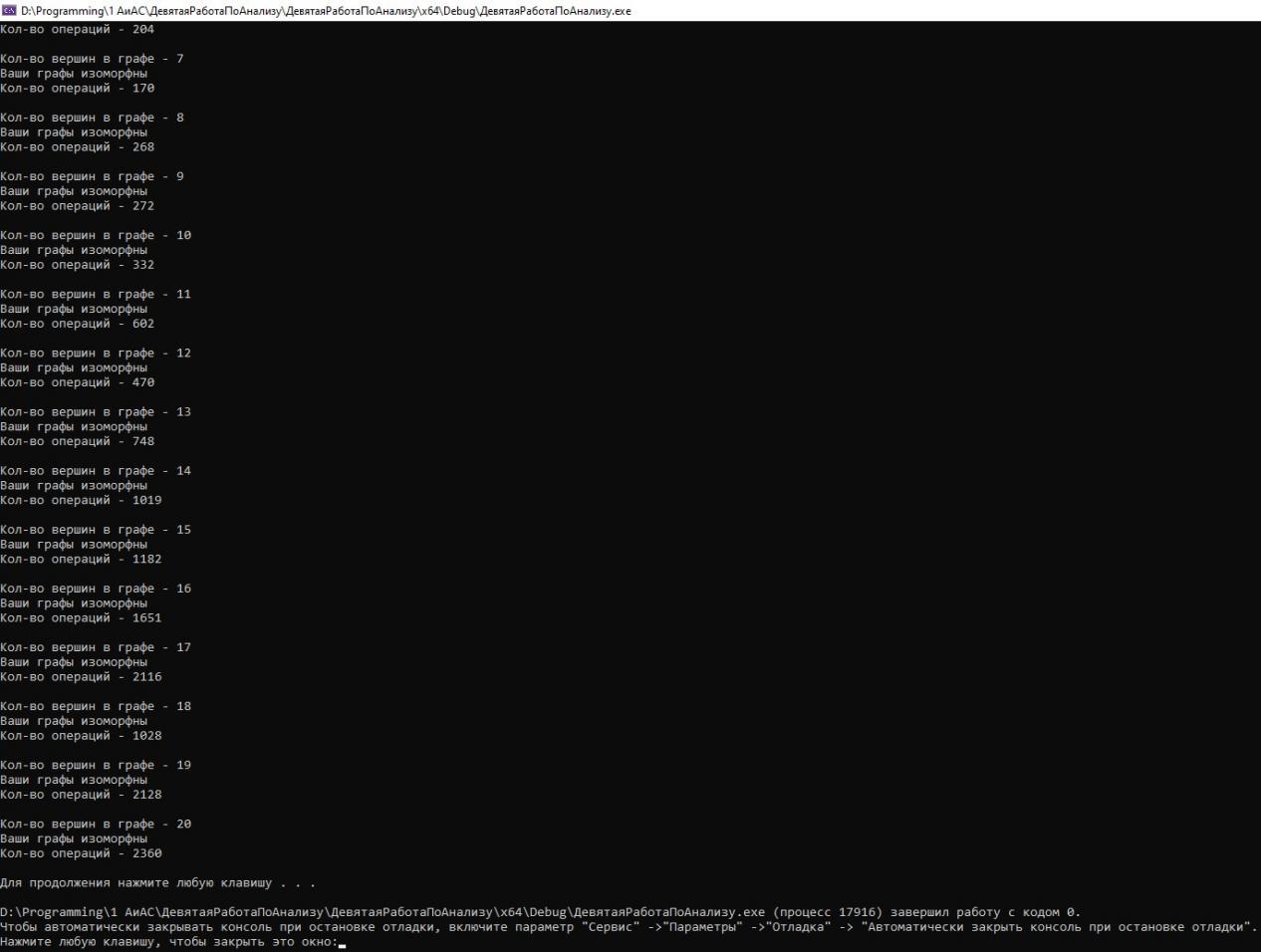
kolop = 0;

}

system("pause");

}

**Пример вывода в консоль:**



**График зависимости и таблица:**

|  |  |
| --- | --- |
| Кол-во вершин | Кол-во операций |
| 1 | 8 |
| 2 | 30 |
| 3 | 66 |
| 4 | 88 |
| 5 | 92 |
| 6 | 299 |
| 7 | 170 |
| 8 | 218 |
| 9 | 272 |
| 10 | 332 |
| 11 | 772 |
| 12 | 766 |
| 13 | 1068 |
| 14 | 1148 |
| 15 | 998 |
| 16 | 1553 |
| 17 | 1596 |
| 18 | 1248 |
| 19 | 1954 |
| 20 | 2238 |

Решим систему уравнений методом неопределённых коэффицентов, и найдём наши значения:

, где *n*> 0

Получаем следующие значения: a = , b = , c = .

В итоге, наша формула имеет следующий вид: , где *n*> 0