Министерство науки и высшего образования Российской Федерации

Пензенский государственный университет

Кафедра «Вычислительная техника»

**ОТЧЕТ**

по лабораторной работе №9

по курсу «Логика и основы алгоритмизации в инженерных задачах»

на тему: «Поиск расстояний в графе»

Выполнили студенты

группы 21ВВ4:

Савкин В.В.

Савостин Е.Е.

Приняли:

Юрова О. В.

Акифьев И.В.

Пенза 2022

**Задание 1**

1. Сгенерируйте (используя генератор случайных чисел) матрицу смежности для неориентированного графа *G*. Выведите матрицу на экран.
2. Для сгенерированного графа осуществите процедуру поиска расстояний, реализованную в соответствии с приведенным выше описанием. При  реализации алгоритма в качестве очереди используйте класс **queue** из стандартной библиотеки С++.

Листинг:

#include <stdio.h>

#include <time.h>

#include <stdlib.h>

#include <locale.h>

#include <queue>

using namespace std;

queue<int>q;

void BFSD(int s, int n, int\* dis, int\*\* G) {

q.push(s);

dis[s] = 0;

while (!q.empty()) {

s = q.front();

q.pop();

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

{

if (G[s][i] == 1 && dis[i] == -1)

{

q.push(i);

dis[i] = dis[s] + 1;

printf("%d(%d) ", dis[i], i);

}

}

}

}

int main() {

int n, m, i = 0;

srand(time(NULL));

setlocale(LC\_ALL, "Russian");

printf("Введите количество вершин для матрицы: ");

scanf\_s("%d", &n);

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

int\* dis = new int[n];

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

{

M[i] = new int[n];

dis[i] = -1;

}

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

{

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

{

if (i == j)

{

M[i][j] = 0;

}

else if (i != j)

{

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

M[j][i] = M[i][j];

}

}

}

printf("Матрица смежности:\n");

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

{

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

{

printf("%d ", M[i][j]);

}

printf("\n");

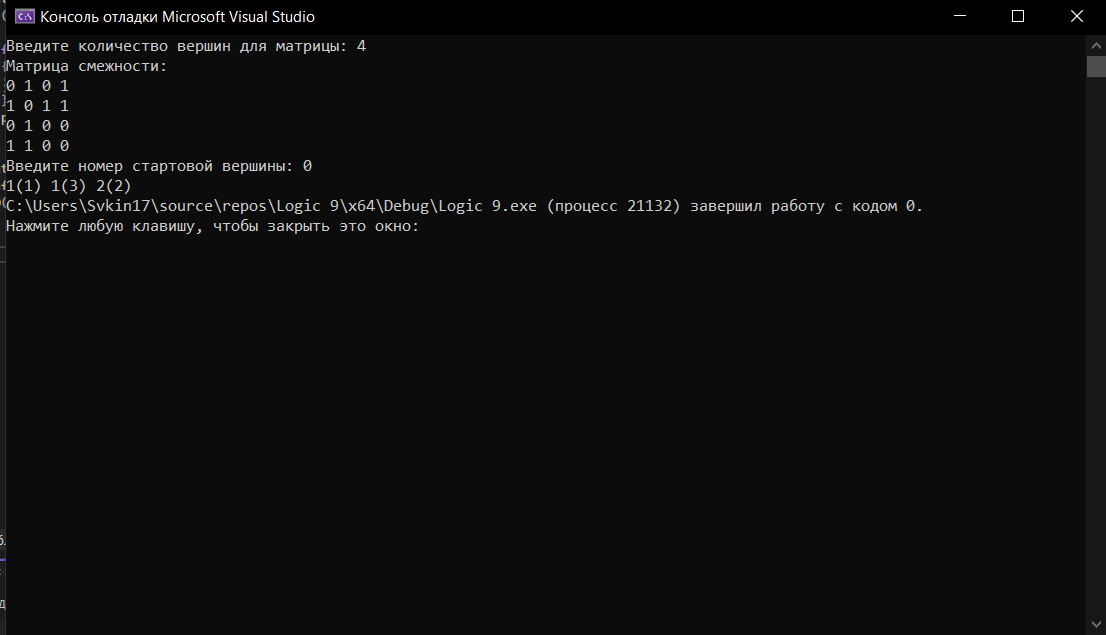
}

printf("Введите номер стартовой вершины: ");

scanf\_s("%d", &m);

BFSD(m, n, &dis[0], &(&M)[0][0]);

}



Вывод: Мы научились реализовывать алгоритм нахождения расстояний в графе.