Министерство образования Российской федерации

Пензенский государственный университет

Кафедра «Вычислительная техника»

**ОТЧЕТ**

по лабораторной работе №9

по курсу «Логика и основы алгоритмизации в ИЗ»

на тему «Поиск расстояний в графе»

Выполнили

студенты группы 21ВВ1.3:

Лукина Екатерина

Тарасов Никита

Приняли:

Юрова О.В.,

Митрохин М.А.

Пенза 2022

**Название**

Поиск расстояний в графе

**Цель работы**

Изучение алгоритма поиска расстояний в графе.

**Лабораторное задание**

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++.

**Ход работы**

**Программа на языке Си**

**Задание №1 и №2:**

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

#include <locale.h>

#include <iostream>

#include <queue>

using namespace std;

int BFSD(int s, int\* vis, int\*\* a, int size) {

queue <int> Q;

vis[s] = 0;

Q.push(s);

while (!Q.empty()) {

s = Q.front();

Q.pop();

}

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

if (a[s][i] == 1 && vis[i] == 0) {

Q.push(i);

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

printf("%d ", i);

}

}

return(0);

}

int main() {

int i, j, \*\* a = NULL, n, \* vis = NULL, start;

printf("Input size of array: ");

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

printf("\n");

a = (int\*\*)malloc(n \* sizeof(int\*));

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

a[i] = (int\*)malloc(n \* sizeof(int));

}

vis = (int\*)malloc(n \* sizeof(int));

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

vis[i] = 0;

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

}

printf("\n\n");

i = 0;

j = 0;

while (i < n) {

j = 0;

while (j < n) {

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

a[j][i] = a[i][j];

if (i == j) {

a[i][j] = 0;

}

j++;

}

i++;

}

i = 0;

j = 0;

while (i < n) {

j = 0;

while (j < n) {

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

j++;

}

printf("\n");

i++;

}

while (1) {

printf("\nInput number: ");

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

if (start == 123) {

break;

}

if (start < n) {

BFSD(start, vis, a, n);

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

vis[i] = 0;

}

}

else {

printf("This number is not available, try again");

}

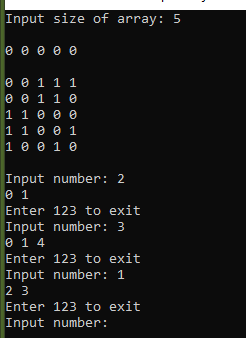
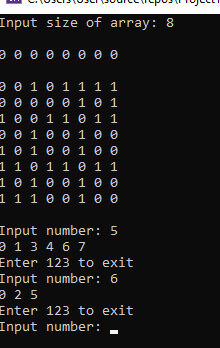
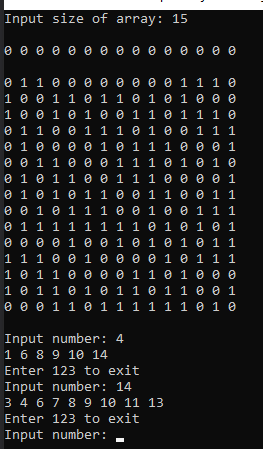
printf("\nEnter 123 to exit");

}

}

**Некоторые выводы, доказывающие правильность составленной программы**

**Задание №1 и 2:**

**Вывод**

В ходе работы был изучен и реализован алгоритм поиска расстояний в графе.