Министерство образования Российской Федерации

Пензенский государственный университет

Кафедра «Вычислительная техника»

**ОТЧЕТ**

по лабораторной работе №4

по курсу «Логика и основы алгоритмизации в инженерных задачах»

на тему «Обход графа в глубину»

Выполнил: студент группы 19ВВ2

Гуськова Н.Д.

Принял:

Митрохин М.А.

Юрова О.В

Пенза 2020

**Цель работы:** составить программу выполнения алгоритма обхода графа в глубину на ЭВМ.

**Ход работы:**

**Задание 1:**

1. Сгенерировали (используя генератор случайных чисел) матрицу *M* смежности неориентированного помеченного графа *G*. Вывели сгенерированную матрицу на экран.

**Задание 2**:

1. Для матричной формы представления графов выполнили обход в глубину.

Результат выполнения операции вывели на экран.

**Листинг:**

Header.h

#pragma once

int pr\_adj\_matrix(char\*\* matrix, int size);

char\*\* gen\_adj\_matrix(int size);

void POG(char\*\* matrix, int size);

main.cpp

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include <conio.h>

#include <stdlib.h>

#include <windows.h>

#include "Header.h"

int main() {

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

int size;

printf("Введите размер матрицы: ");

scanf("%d", &size);

char\*\* adj = gen\_adj\_matrix(size);

pr\_adj\_matrix(adj, size);

POG(adj, size);

\_getch();

}

adjmatrixgen.cpp

#include <time.h>

#include <stdlib.h>

char\*\* gen\_adj\_matrix(int size) {

srand(time(0));

char\*\* matrix = (char\*\*)calloc(size, sizeof(char\*));

if (matrix == NULL)

return NULL;

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

matrix[i] = (char\*)calloc(size, sizeof(char));

if (matrix[i] == NULL)

return NULL;

}

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

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

if (i == j) {

matrix[i][j] = 0;

}

else {

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

matrix[j][i] = matrix[i][j];

}

}

}

return matrix;

}

DFS.cpp

#include <stdlib.h>

#include <stdio.h>

void DFS(char\*\* matrix, int v, char\* visited, int size) {

visited[v] = 1;

printf("%d", v);

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

if ((visited[i] == 0) && (matrix[v][i] == 1))

{

DFS(matrix, i, visited, size);

}

}

}

void POG(char\*\* matrix, int size) {

char\* visited = (char\*)calloc(size, sizeof(char));

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

visited[i] = 0;

}

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

{

if (visited[v] == 0) {

DFS(matrix, v, visited, size);

}

}

}

pr\_output.cpp

#include <stdio.h>

int pr\_adj\_matrix(char\*\* matrix, int size) {

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

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

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

}

printf("\n");

}

return 0;

}

Результаты работы программы:

**Вывод:** составил программу обхода графа в глбину на ЭВМ.