Министерство образования Российской Федерации

Пензенский государственный университет

Кафедра «Вычислительная техника»

**Отчёт**

по лабораторной работе №4

по курсу «Логика и основы алгоритмизации в инженерных задачах»

на тему «Обход графа в глубину»

Выполнил:

 студент группы 20ВВ2

Лазутин Д.Д.

 Принял:

 к.т.н., доцент Юрова О.В.

Пенза 2020

### **Задание 1**

1. Сгенерируйте (используя генератор случайных чисел) две матрицу смежности для неориентированного графа *G*. Выведите сгенерированные матрицы на экран.
2. Для сгенерированного графа осуществите процедуру обхода в глубину, реализованную в соответствии с приведенным выше описанием.

**3.**\* Реализуйте процедуру обхода в глубину для графа, представленного списками смежности. (В разработке)

**Описание метода решения:**

Объявлены переменные: матрица G, размер матрицы S, массив посещённых вершин графа Visited, счётчики i и j, номер стартовой вершины N и датчик повторного посещения вершины N, чтобы вовремя прекратить процедуру обхода.  
Создаём и заполняем матрицу смежности, выводим её на экран, потом вводим номер стартовой вершины (т.е. откуда начнётся обход). Далее выполняем обход графа в глубину посредством вызова рекурсивной функции DFS до тех пор, пока все вершины не будут посещены (т.е. все элементы массива Visited должны стать равными 1).

**Листинг** LZ\_4.c

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include <time.h>

#include <stdlib.h>

#include <locale.h>

void DFS(int v, int \*\*M, int \*Vis, int Size)

{

int i;

Vis[v]=1;printf("%3d", v+1);

for(i=0;i<Size;i++)

if((M[v][i]==1)&&(Vis[i]==0))

DFS(i, M, Vis, Size);

}

void main()

{

int \*\*G=NULL, i=0, j=0, S=0, \*Visited=NULL;

setlocale(LC\_ALL, "Rus");

srand(time(NULL));

printf("Введите размерность матрицы смежности: ");

scanf("%d", &S);

while(S<=0)

{

printf("Некорректно введённое число! Введите целое положительное число: ");

scanf("%d", &S);

}

G=(int\*\*)malloc(S\*sizeof(int));

Visited=(int\*)malloc(S\*sizeof(int));

for(i=0;i<S;i++)

Visited[i]=0;

for(i=0;i<S;i++)

{

G[i]=(int\*)malloc(S\*sizeof(int));

for(j=0;j<S;j++)

{

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

if(G[i][i]!=0)

G[i][j]=0;

}

}

for(i=0;i<S;i++)

{

printf("\n%4d|", i+1);

for(j=0;j<S;j++)

{

G[j][i]=G[i][j];

printf("%2d", G[i][j]);

}

}

printf("\n\nПорядок обхода графа в глубину: ");

for(i=0;Visited[i]!=1;i++)

DFS(i, G, Visited, S);

printf("\n");

free(Visited);

free(G);

}

### **Задание 2\***

1. Для матричной формы представления графов выполните преобразование рекурсивной реализации обхода графа к не рекурсивной. (В разработке)

**Результаты работы программ**

Результаты работы программы для 1+2-го заданий представлены на рис.1-2.

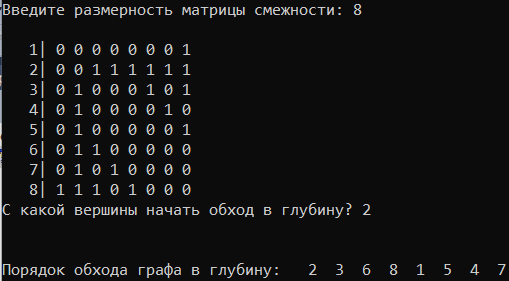


Рис.1.

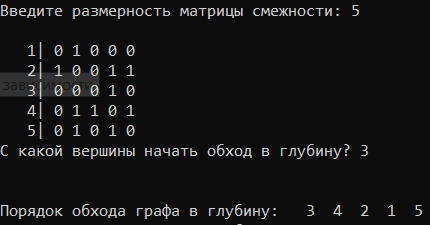


Рис.2.