Министерство науки и высшего образования Российской Федерации

Пензенский государственный университет

Кафедра «Вычислительная техника»

**ОТЧЁТ**

по лабораторной работе №5

по курсу «Логика и основы алгоритмизации в ИЗ»

## на тему «Определение характеристик графов»

Выполнил:

студент групп 22ВВВ2

Дасаев Т.З.

Приняли:

д.т.н. профессор Митрохин М.А.

к.э.н. доцент Акифьев И.В.

Пенза 2023

**Название**

Определение характеристик графов

**Цель работы**

Сгенерировать матрицу смежности для неориентированного графа G. Вывести ее на экран, найти изолированные, концевые и доминирующие вершины.

**Лабораторное задание**

**Задание**

### Задание 1

1. Сгенерируйте (используя генератор случайных чисел) матрицу смежности для неориентированного графа *G*. Выведите матрицу на экран.
2. Определите размер графа *G*, используя матрицу смежности графа.
3. Найдите изолированные, концевые и доминирующие вершины.

### Задание 2\*

1. Постройте для графа G матрицу инцидентности.
2. Определите размер графа *G*, используя матрицу инцидентности графа.
3. Найдите изолированные, концевые и доминирующие вершины.

**Листинг**

#include <stdio.h>

#include <time.h>

#include <stdlib.h>

#include <locale.h>

#define m 6

int main ()

{

setlocale(LC\_ALL, "rus");

int A[m][m] = { 0 };

int M = 0, I = 0, D = 0, K = 0, N = 0, P = 0;

srand(time(NULL));

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

{

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

{

int k = rand() % 2;

A[i][j] = k;

A[j][i] = k;

A[i][i] = 0;

}

}

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

{

int n = 0;

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

{

if (A[i][j] == 1)

n++;

}

if (n > N)

N = n;

}

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

{

int n = 0;

int k = 0;

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

{

if (A[i][j] == 0)

{

k++;

}

if (A[i][j] == 1)

{

M++;

n++;

}

}

if (k == m)

I++;

if (k == m - 1)

K++;

if (n == N)

D++;

}

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

{

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

{

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

}

printf("\n");

}

printf("Размер матрицы смежности - %d\n", M / 2);

printf("Изолированные вершины - %d\n", I);

printf("Концевые вершины - %d\n", K);

printf("Доминирующие вершины - %d\n", D);

I = 0;

K = 0;

D = 0;

//\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

int\*\* k = (int\*\*)malloc(m \* sizeof(int\*));

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

{

k[i] = (int\*)malloc(M / 2 \* sizeof(int));

}

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

{

for (int j = 0; j < M / 2; j++)

{

k[i][j] = 0;

}

}

int count = 0;

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

{

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

{

if (j > i)

{

if (A[i][j] == 1)

{

k[i][count] = 1;

k[j][count] = 1;

count++;

}

}

}

}

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

{

for (int j = 0; j < M / 2; j++)

{

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

}

printf("\n");

}

int O = 0;

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

{

int d = 0;

for (int j = 0; j < M / 2; j++)

{

if (k[i][j] == 1)

d++;

}

if (d > O)

O = d;

}

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

{

int d = 0;

int y = 0;

for (int j = 0; j < M / 2; j++)

{

if (k[i][j] == 0)

d++;

if (k[i][j] == 1)

y++;

}

if (d == M / 2)

I++;

if (d == M / 2 - 1)

K++;

if (M / 2 - d > O)

O = M / 2 - d;

if (y == O)

D++;

}

printf("Размер матрицы инцидентности - %d\n", M / 2);

printf("Изолированные вершины - %d\n", I);

printf("Концевые вершины - %d\n", K);

printf("Доминирующие вершины - %d\n", D);

return 0;

}

**Результат работы программы**

Результат работы программы показан на рисунках.

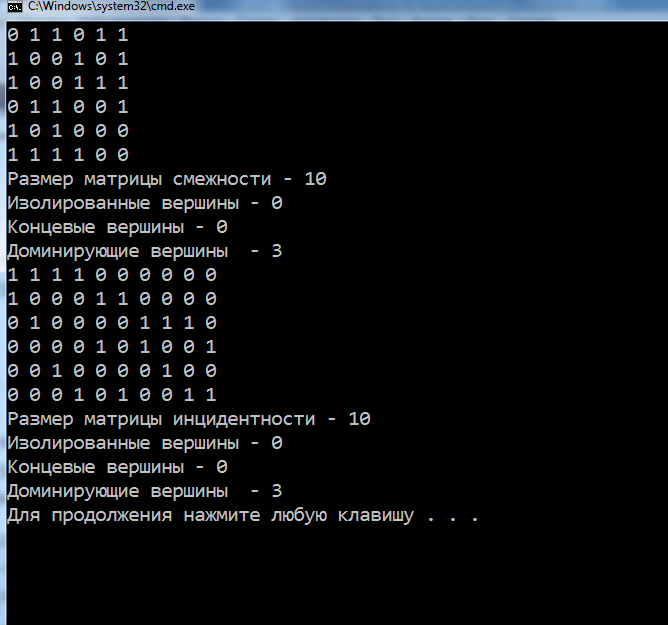


Рис. 1

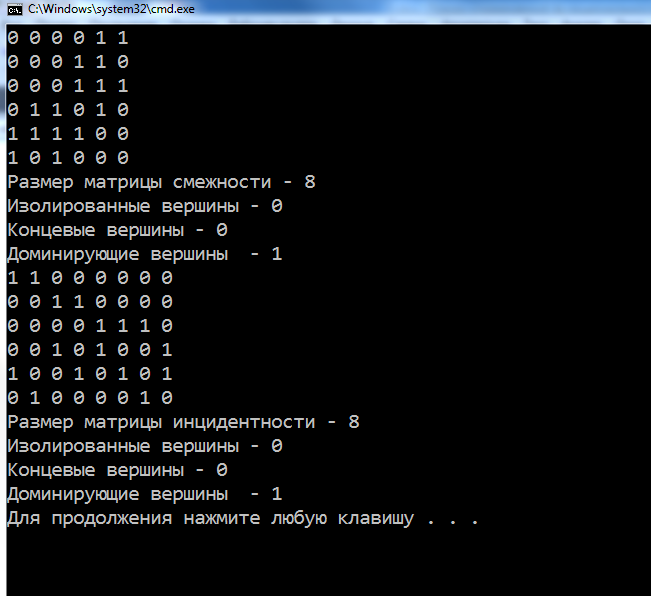


Рис. 2

**Вывод:** в ходе выполнения лаб. работы мы научились создавать матрицу смежности и инцидентности для неориентированного графа G, а также находить изолированные, концевые и доминирующие вершины.

Ссылка на репозиторий: https://github.com/t1muurr/LiAO\_l\_5.git