Министерство науки и высшего образования Российской Федерации

Пензенский государственный университет

Кафедра "Вычислительная техника"

Отчет

по лабораторной работе №5

по курсу “Л и ОА в ИЗ”

на тему “Определение характеристик графов”

Выполнил студенты группы 22ВВС1:

Агапов И.Е.

Братчиков А.А.

Приняли:

Юрова О.В.

Акифьев И.В.

Пенза 2023

### Лабораторное 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**

#define \_CRT\_SECURE\_NO\_WARNINGS

#include <stdio.h>

#include <time.h>

#include <stdlib.h>

#include <iostream>

int main() {

srand(time(NULL));

int size = 0;

int w;

int max = 0;

int min = 0;

scanf("%d", &size);

w = size;

int\*\* mas;

mas = (int\*\*)malloc(sizeof(int\*) \* size);

int size1 = size;

int i = 0;

int j = 0;

int s = 0;

setlocale(LC\_ALL, "");

while (size1 > i) {

j = 0;

mas[i] = (int\*)malloc(sizeof(int) \* w);

while (w > j) {

mas[i][j] = 0;

j++;

}

i++;

}

i = 0;

j = 0;

while (size1 > i) {

j = i;

while (w > j) {

if (i == j) {

mas[i][j] = 0;

}

else {

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

mas[j][i] = mas[i][j];

}

printf("%d\t", mas[i][j]);

j++;

}

printf("\n");

i++;

}

int a = 1;

int b = 1;

int c = 1;

int\* iso = (int\*)malloc(sizeof(int) \* size);

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

iso[i] = -1;

}

iso[0] = 0;

int\* con = (int\*)malloc(sizeof(int) \* size);

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

con[i] = -1;

}

con[0] = 0;

int\* dom = (int\*)malloc(sizeof(int) \* size);

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

dom[i] = -1;

}

dom[0] = 0;

i=0;

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

for (int n = 0; n < w; n++) {

printf("%d\t", mas[o][n]);

s += mas[o][n];

}

printf("| %d", s);

if (s == 0) {

iso[0]++;

iso[b] = o;

b++;

}

else if(s==1){

con[0]++;

con[a] = o;

a++;

}

else if (s == size-1) {

dom[0]++;

dom[c] = o;

c++;

}

i += s;

s = 0;

printf("\n");

}

int razm=i/2;

printf("Размер графа %d\n", razm);

i = 1;

printf("Количество изолированных вершин %d\n", iso[0]);

for (int i = 1; i < size; i++) {

if (iso[i] != -1) {

printf("Вершина - %d\n", iso[i]+1);

}

}

i = 1;

printf("Количество концевых вершин %d\n", con[0]);

for (int i = 1; i < size; i++) {

if (con[i] != -1){

printf("Вершина - %d\n", con[i]+1);

}

}

i = 1;

printf("Количество доминирующих вершин %d\n", dom[0]);

for (int i = 1; i < size; i++) {

if (dom[i] != -1) {

printf("Вершина - %d\n", dom[i]+1);

}

}

int\*\* iden= (int\*\*)malloc(sizeof(int\*) \* size);

i = 0;

j = 0;

while (razm > i) {

j = 0;

iden[i] = (int\*)malloc(sizeof(int) \* w);

while (w > j) {

iden[i][j] = 0;

j++;

}

i++;

}

int\* sum = (int\*)malloc(sizeof(int) \* size);

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

sum[i] = 0;

}

int m=0;

int k=0;

while (razm > m) {

k = m;

while (w >k){

if(mas[m][k]==1){

iden[m][k]=1;

iden[m][m]=1;

}

k++;

}

m++;

}

for (int o = 0; o < razm; o++) {

for (int n = 0; n < w; n++) {

printf("%d\t", iden[o][n]);

sum[n]+=iden[o][n];

}

printf("\n");

}

printf("============================================\n");

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

printf("%d\t", sum[j]);

}

printf("\n");

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

if(sum[j]==1){

printf("Концевая вершина - %d\n", j+1);

}else if(sum[j]==0){

printf("Изолированная вершина - %d\n", j+1);

}else if(sum[j]==w-1){

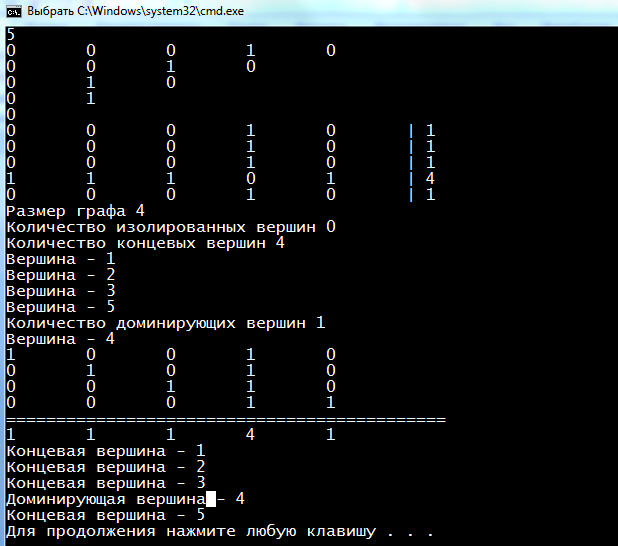
printf("Доминирующая вершина - %d\n", j+1);

}

}

}

**Результаты работ программ**



**Вывод**

### В ходе выполнения лабораторной работы были обобщены знания о графах. Были реализованы генерация матрицы смежности и нахождение размера графа, изолированных, концевых, доминирующих вершин.

### Была построена матрица инцидентности на основе матрицы смежности. Рассчитаны размер графа, кол-во изолированных, концевых, доминирующих вершин.