Министерство науки и высшего образования Российской Федерации

Пензенский государственный университет

Кафедра «Вычислительная техника»

**ОТЧЕТ**

по лабораторной работе №5

по курсу “Логика и основа алгоритмизации в ИЗ.”

на тему “ Определение характеристик графов”

Выполнили студенты группы 21ВВ3:

Тюкалов В.Е.

Чинов Д.Д.

Приняли:

Митрохин М.А., Юрова О.В.

Пенза 2022

**Название**

Определение характеристик графов.

**Цель работы:** выполнить лабораторные указания 1-4 реализуя унарные и бинарные операции над графами.

**Лабораторное задание:**

1. Сгенерируйте (используя генератор случайных чисел) матрицу смежности для неориентированного взвешенного графа G. Выведите матрицу на экран.
2. Определите размер графа G, используя матрицу смежности графа.
3. Найдите изолированные, концевые и доминирующие вершины по матрице смежности.
4. Постройте для графа G матрицу инцидентности.
5. Найдите изолированные, концевые и доминирующие вершины по матрице инцидентности.

**Листинг**

//Выполнили студенты группы 21ВВ3: Тюкалов В.Е. и Чинов Д.Д.

#include<iostream>

#include<stdio.h>

#include<stdlib.h>

#include<time.h>

#include<windows.h>

int main() {

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

const int N = 5;

char c = 0;

int\* countColI,\* countColJ;

int countCol = 0, count = 0;

char Matrix[N][N] = {0};

for (int i = N; i > 0; i--) {

count = count + i;

}

countColI = (int\*)malloc(count \* sizeof(int));

countColJ = (int\*)malloc(count \* sizeof(int));

count = 0;

srand(time(NULL));

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

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

if (i == j) continue;

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

if (Matrix[i][j] == 1) {

countColI[countCol] = i;

countColJ[countCol] = j;

countCol++;

}

Matrix[j][i] = Matrix[i][j];

}

}

printf(" Матрица смежности\n");

printf("\n a b c d e\n");

printf(" \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n");

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

printf(" %c | ", (i + 97));

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

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

}

printf("\n");

}

printf("\n Размер графа: %d\n\n", countCol);

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

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

if (Matrix[i][j] == 1) count++;

}

if (count == 0) printf(" %c - изолированная\n", (i + 97));

else if (count == 1) printf(" %c - конечная\n", (i + 97));

else printf(" %c - доминирующая\n", (i + 97));

count = 0;

}

printf("\n Матрица инцидентности\n");

printf("\n a b c d e\n");

printf(" \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\n");

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

printf(" %c-%c | ", (countColI[i] + 97), (countColJ[i] + 97));

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

if (countColI[i] == j || countColJ[i] == j) printf("1 ");

else printf("0 ");

}

printf("\n");

}

return 0;

}

**Результаты работы программы:**

Результаты работы программы показан на рисунке 1.

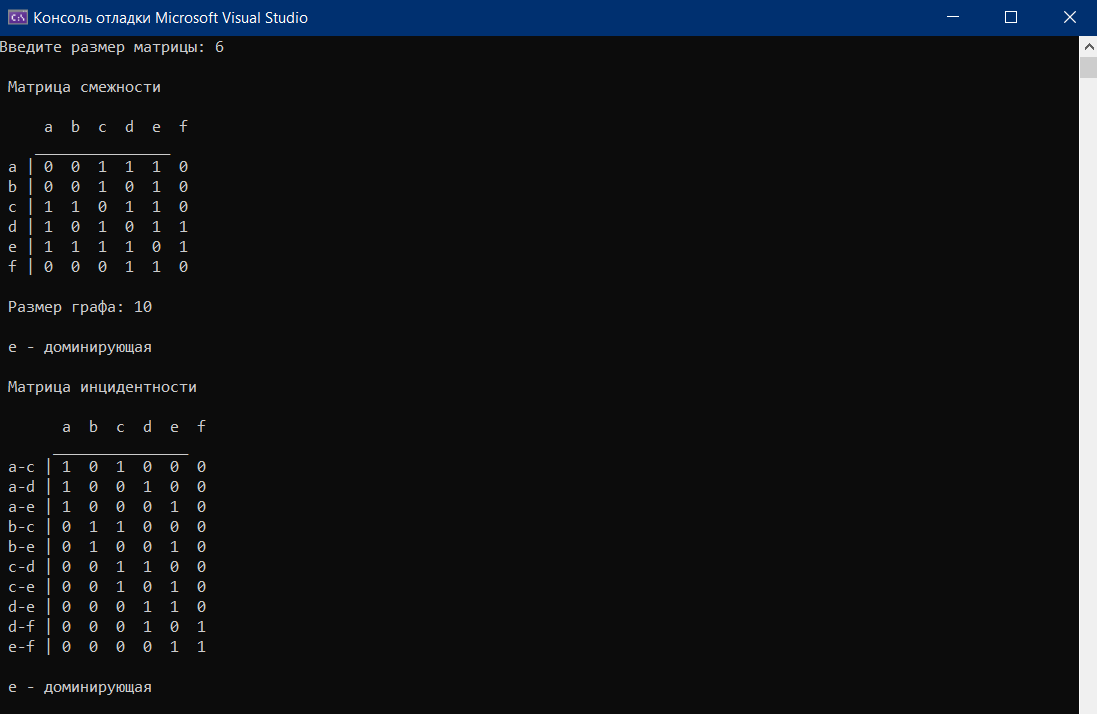


Рис. 1

**Задание 1:**

Была сгенерирована и построена матрица смежности для графа G.

**Задание 2:**

Определили размер графа G.

**Задание 3:**

Найдены изолированные, концевые и доминирующие вершины по матрице смежности.

**Задание 4:**

Была построена матрица инцидентности для графа G.

**Задание 5:**

Найдены изолированные, концевые и доминирующие вершины по матрице инцидентности.

**Вывод:** Выполнив лабораторные задания 1-4, мы научились работать с матрицей смежности: находить вершины, находить размер графа, строить матрицу инцидентности.