МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ

РОССИЙСКОЙ ФЕДЕРАЦИИ

ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Кафедра «Вычислительной техники»

**ОТЧЁТ**

по лабораторной работе №5

по курсу «Л и ОА в ИЗ»

на тему «Определение характеристик графов»

Выполнили:

Студенты группы 22ВВС1

Костин Максим

Макеева Дарья

Приняли:

Акифьев И.В.

Юрова О.В.

Пенза 2023

**Лабораторноезадание*:***

**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 <conio.h>

#include <locale.h>

#include <stdlib.h>

#include <time.h>

int main() {

setlocale(LC\_ALL, "Rus");

srand(time(NULL));

int n = 0;

printf("Введите количество вершин: ");

scanf\_s("%d", &n);

int\*\* matr = (int\*\*)malloc(sizeof(int\*) \* n);

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

matr[i] = (int\*)malloc(sizeof(int) \* n);

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

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

matr[i][j] = 0;

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

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

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

matr[j][i] = matr[i][j];

if (i == j)

matr[i][j] = 0;

}

}

printf("\n Матрица:\n№ ");

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

printf(" %d", i + 1);

}

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

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

printf("\_\_");

}

printf("\n");

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

{

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

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

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

printf("\n");

}

}

printf("------------------------------------------\n");

int kr = 0;

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

for (int j = i; j < n; j++)

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

kr += 1;

}

}

printf("\nРазмер графа = %d", kr);

int domin = 0, izol = 0, k = 0, kv = 0;

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

domin = 0;

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

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

domin += 1;

}

if (domin == n - 1) {

printf("\nДоминирующая вершина - №%d", i + 1);

k++;

}

if (domin == 0) {

printf("\nИзолированная вершина - №%d", i + 1);

izol++;

}

if (domin == 1) {

printf("\nКонцевая вершина - №%d", i + 1);

kv++;

}

}

if (kv == 0)

printf("\nКонцевых вершин нет");

if (k == 0)

printf("\nДоминирующих вершин нет");

if (izol == 0)

printf("\nИзолированных вершин нет");

printf("\n------------------------------------------\n");

int\*\* matr1 = (int\*\*)malloc(sizeof(int\*) \* n);

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

matr1[i] = (int\*)malloc(sizeof(int) \* kr);

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

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

matr1[i][j] = 0;

int in = 0, in1 = 0, in2 = 0;

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

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

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

matr1[i][in] = 1;

matr1[j][in] = 1;

in++;

}

}

}

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

free(matr[i]);

}

free(matr);

printf("\n Матрица:\n№ ");

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

printf(" %d", i + 1);

}

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

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

printf("\_\_");

}

printf("\n");

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

{

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

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

printf("%d ", matr1[i][j]);

printf("\n");

}

}

printf("------------------------------------------\n");

domin = 0, izol = 0, k = 0, kv = 0;

int k1 = 0;

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

domin = 0;

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

if (matr1[i][j] == 1) {

domin += 1;

}

}

if (domin == n - 1) {

printf("\nДоминирующая вершина - №%d", k1 + 1);

k++;

}

if (domin == 0) {

printf("\nИзолированная вершина - №%d", k1 + 1);

izol++;

}

if (domin == 1) {

printf("\nКонцевая вершина - №%d", k1 + 1);

kv++;

}

k1++;

}

if (kv == 0)

printf("\nКонцевых вершин нет");

if (k == 0)

printf("\nДоминирующих вершин нет");

if (izol == 0)

printf("\nИзолированных вершин нет");

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

free(matr1[i]);

}

free(matr1);

\_getch();

}

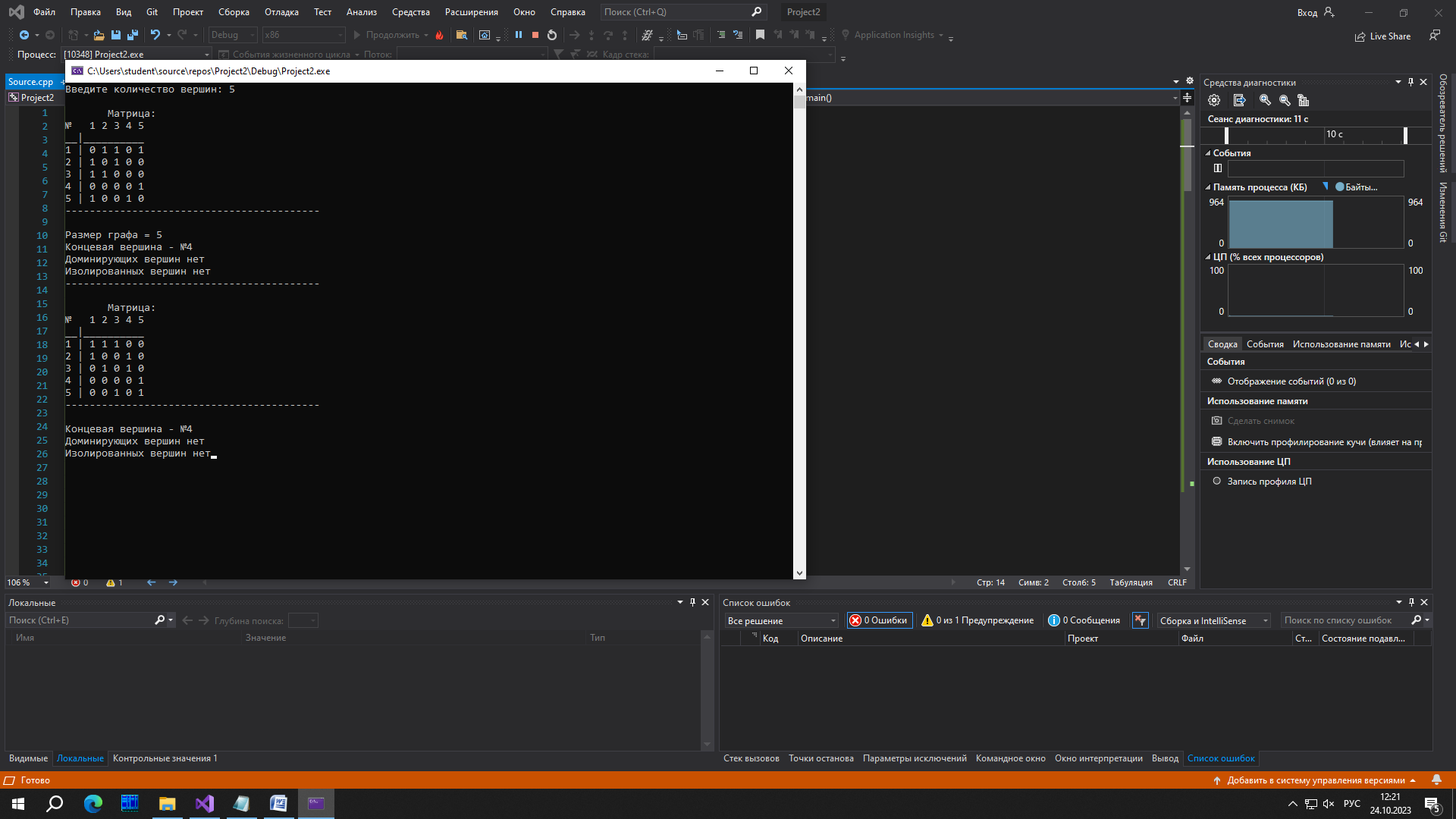
# Пояснительный текст к программе:

int\*\* matr = (int\*\*)malloc(sizeof(int\*) \* n);

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

matr[i] = (int\*)malloc(sizeof(int) \* n); — создание двумерного динамического массива.

**Результат работы программы**

****

**Вывод**

В ходе выполнения лабораторной работы научились определять характеристики графа и сделали матрицу инцидентности на основе матрицы смежности.