|  |
| --- |
| Министерство образования Российской Федерации  Пензенский государственный университет  Кафедра «Вычислительная техника» |
| Отчет  по лабораторной работе №6  по курсу «Логика и основы алгоритмизации в инженерных задачах»  на тему «Поиск расстояний в графе» |
|  |
|  |
| Выполнил студент группы 19ВВ3:  Литвинов А.Ю.  Принял:  Митрохин М. А. |
| Пенза  2020 |

**Цель работы**

Реализовать алгоритм поиска расстояний в графе, используя алгоритм обхода в ширину.

**Листинг программы**

#include <stdio.h>

#include <time.h>

#include <windows.h>

#include <conio.h>

#include <queue>

#include <iostream>

#pragma warning (disable : 4996)

int\* DIST;

int\*\* arr;

int\* memory;

void BFSD(int v, int a) {

std::queue <int> qu;

qu.push(v);

DIST[v] = 0;

while (!qu.empty()) {

v = qu.front();

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

qu.pop();

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

if ((arr[v][i] == 1) && (DIST[i] == -1)) {

qu.push(i);

DIST[i] = DIST[v] + 1;

}

}

}

}

void main()

{

int begin;

int n;

int min=1000;

SetConsoleCP(1251);

SetConsoleOutputCP(1251);

srand(time(NULL));

printf("Введите порядок массива - ");

scanf("%d", &n);

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

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

DIST[i] = -1;

}

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

for (int i = 0; i < n; i++) arr[i] = (int\*)malloc(n \* sizeof(int));

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

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

if (j > i) {

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

arr[j][i] = arr[i][j];

}

else if (i == j) arr[i][j] = 0;

}

}

printf(" ");

for (int k = 1; k <= n; k++) printf(" %d", k);

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

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

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

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

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

}

printf("\n");

}

printf("Введите вершину с которой хотите начать обход - ");

scanf("%d", &begin);

printf("Посещенные вершины: ");

BFSD(begin - 1, n);

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

printf("| Вершины |");

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

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

}

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

printf("| Расстояние |");

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

printf("| %d |", DIST[i]);

}

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

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

if ((min>DIST[i]) && (DIST[i]!=0)) {

min = DIST[i];

}

}

printf(" Кратчайшее расстояние - %d , у вершины:", min);

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

if (DIST[i]==min) {

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

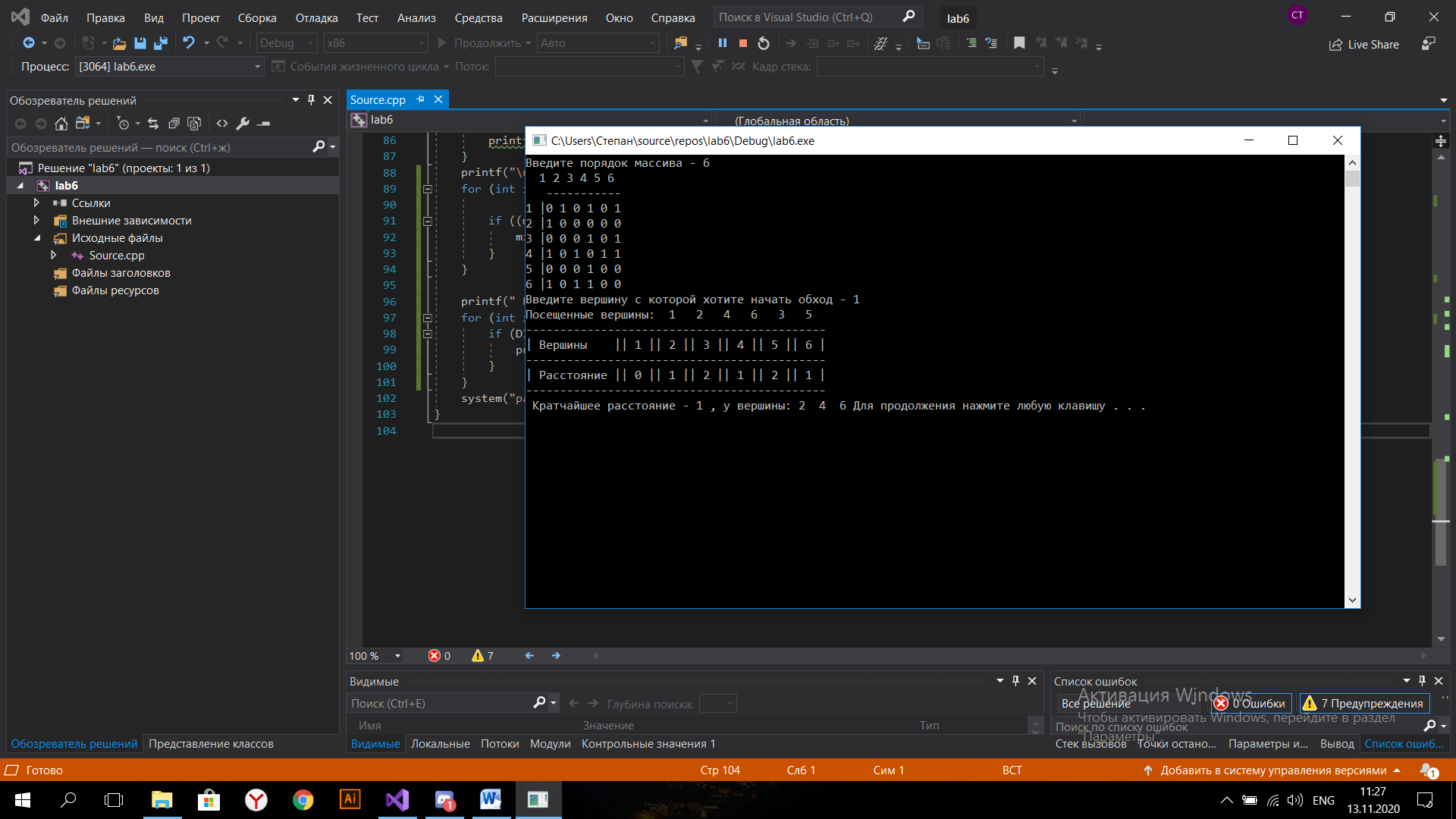
}

}

system("pause");

}

### **Результат работы программы:**



**Вывод:** Реализовал алгоритм поиска расстояний в графе, используя алгоритм обхода в ширину.