Министерство образования Российской Федерации

Пензенский государственный университет

Кафедра «Вычислительная техника»

**ОТЧЁТ**

по лабораторной работе №5

по курсу «Логика и основы алгоритмизации в инженерных задачах»

на тему «Обход графа в ширину»

Выполнили:

студенты группы 20ВВ2

Киреев Б.П.

Верховский М.В.

Лукин В.Д.

Приняли:

д.т.н. Митрохин М.А.

к.т.н. доцент Юрова О.В.

Пенза 2021

Название: Обход графа в ширину.

Цель работы: Научиться осуществлять обход графа в ширину.

Лабораторное задание:

**Задание 1:**

1. Сгенерируйте (используя генератор случайных чисел) матрицу

смежности для неориентированного графа G. Выведите матрицу на экран.

2. Для сгенерированного графа осуществите процедуру обхода в ширину,

реализованную в соответствии с приведенным выше описанием. При

реализации алгоритма в качестве очереди используйте класс queue из

стандартной библиотеки С++.

**Задание 1:**

Листинг:

Файл Lab5.cpp

#include <time.h>

#include <malloc.h>

#include <locale.h>

#include <iostream>

#include <queue>

#include <ctime>

#include <conio.h>

using namespace std;

int i, num = 0;

int\* vis = NULL, n, \*\* p = NULL;

int pogr = 0;

void BFS(int\*\* p, int\* vis, int n, int num) {

queue <int> q;

int i;

q.push(num);

vis[num] = 1;

while (!q.empty()) {

num = q.front();

q.pop();

cout << num + 1 << " ";

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

if (vis[i] == 0 && p[num][i] == 1) {

q.push(i);

vis[i] = 1;

}

}

}

}

int main() {

setlocale(LC\_ALL, "Rus");

srand(time(0)); //по-настоящему случайная генерация массива

cout << "Введите размер матрицы: ";

cin >> n;

cout << "Матрица: \n";

vis = (int\*)calloc(n, 3);

p = (int\*\*)calloc(n, 3);

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

p[i] = (int\*)calloc(n, 3);

}

cout.precision(3 \* n);

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

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

vis[i] = 0;

}

for (int i = 0; i < n; i++) { //Генерация матрицы смежности

for (int m = 0 + pogr; m < n; m++) {

p[i][m] = 0 + rand() % 2;

p[m][i] = p[i][m];

p[i][i] = 0;

}

pogr++;

}

for (int i = 0; i < n; i++) { //Отрисовка матрицы смежности

for (int m = 0; m < n; m++) { cout << p[i][m] << " "; }

cout << endl;

}

cout << "Вершины: " << endl;

BFS(p, vis, n, num);

return 0;

}

Пояснительный текст к программе:

Данная программа осуществляет обход графа в ширину. Работа алгоритма продолжается до тех пор, пока существуют непосещённые вершины.

Результаты работы программы:

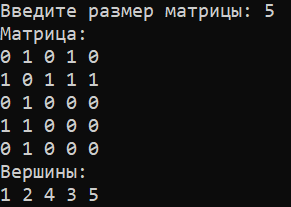


Рисунок №1 – Результат работы программы.

Вывод: При выполнении данной лабораторной работы мы смогли научиться осуществлять обход графа в ширину.