МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ  
ПЕНЗЕНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
Кафедра «Вычислительной техники»

**ОТЧЁТ**

по лабораторной работе №9  
по курсу «Поиск расстояний в графе»  
на тему «Обход графа в ширину»

Выполнил:

Студент группы 23ВВВ2

Пичаев И. А.

Чупраков С. В.

Приняли:

Митрохин М. А.  
Юрова О.В.

Пенза 2024

**Цель работы**

Приобрести навыки программирования и работы алгоритмов нахождения расстояния до других вершин графа различными способами.

**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 поиска расстояний, реализованную в соответствии с приведенным выше описанием. При реализации алгоритма в качестве очереди используйте класс **queue** из стандартной библиотеки С++.

**3.**\* Реализуйте процедуру поиска расстояний для графа, представленного списками смежности.

### **Задание 2\***

1. Реализуйте процедуру поиска расстояний на основе обхода в глубину.
2. Реализуйте процедуру поиска расстояний на основе обхода в глубину для графа, представленного списками смежности.
3. Оцените время работы реализаций алгоритмов поиска расстояний на основе обхода в глубину и обхода в ширину для графов разных порядков.

**Ход работы**

### **Задание 1.1-2**

Программа реализует работу с графами, генерируя случайную матрицу смежности для неориентированного графа и выполняя поиск в ширину (BFS) для нахождения кратчайших расстояний от заданной вершины до всех остальных. Она включает в себя генерацию графа, вывод его представления в виде матрицы смежности и вычисление расстояний с помощью алгоритма BFS:

Инициализация: Вектор расстояний (DIST) инициализируется значениями -1, что означает, что вершины еще не посещены. Начальная вершина (v) помещается в очередь и её расстояние до самой себя устанавливается как 0.

Основной цикл: Пока в очереди есть вершины:

Извлекается вершина из очереди.

Для каждого соседа этой вершины (по матрице смежности) проверяется, был ли он уже посещен. Если не был, то он добавляется в очередь, и его расстояние обновляется как расстояние текущей вершины плюс 1.

Результат: После завершения обхода все элементы вектора DIST содержат кратчайшие расстояния от исходной вершины до всех других, если такие расстояния существуют.

### **Задание 1.3**

Программа предназначена для работы с неориентированным графом. Она генерирует случайный граф с заданным количеством вершин, используя списки смежности, и выполняет поиск в ширину (BFS) от заданной вершины, чтобы вычислить кратчайшие расстояния до всех остальных вершин графа.

Краткое описание работы функции BFS:

Инициализация:

Функция принимает граф в виде списка смежности (adjList), количество вершин (N), исходную вершину (v), и вектор для хранения расстояний от исходной вершины (DIST).

Вектор DIST инициализируется значениями -1, что означает, что вершины ещё не посещены.

Очередь:

Для обхода графа используется очередь (std::queue<int>). Начальная вершина добавляется в очередь, а её расстояние в векторе DIST устанавливается в 0.

Поиск в ширину:

Пока очередь не пуста, берётся очередная вершина, и для каждого её соседа, если он ещё не был посещён (его расстояние равно -1), он добавляется в очередь, а его расстояние обновляется как расстояние текущей вершины + 1.

Результат:

После завершения работы BFS вектор DIST содержит кратчайшие расстояния от начальной вершины до всех остальных вершин графа. Вершины, которые недостижимы из начальной, остаются со значением -1.

### **Задание 2.1-2**

Программа генерирует случайный неориентированный граф, представленный списком смежности, и находит расстояния от указанной вершины до всех остальных с использованием алгоритма обхода в глубину (DFS). Основная цель — продемонстрировать работу алгоритма DFS для поиска кратчайших расстояний в графе и измерить время выполнения этого алгоритма.

Краткое описание работы функции DFS:

Инициализация: Для каждой вершины поддерживается вектор расстояний, где начальные значения установлены в -1 (означает, что вершина ещё не посещена).

Обход графа: Функция DFS вызывается для каждой вершины. Если вершина ещё не посещена, она помечается как посещённая и расстояние от исходной вершины обновляется.

Рекурсия: Функция рекурсивно вызывает себя для всех соседей текущей вершины, увеличивая расстояние на 1 для каждой следующей вершины.

Возврат: После того как все соседи текущей вершины обработаны, функция возвращается к предыдущей вершине и продолжает обход.

Таким образом, DFS обходит все достижимые вершины и находит расстояния от начальной вершины до всех других, если они связаны.

### **Задание 2.3**

### Программа предназначена для работы с графами, представленными списками смежности, и выполнения двух алгоритмов поиска кратчайших расстояний от заданной исходной вершины: поиск в ширину (BFS) и поиск в глубину (DFS). Программа генерирует случайный неориентированный граф с указанным количеством вершин, выполняет поиск в ширину и глубину, а затем выводит расстояния от исходной вершины до всех остальных и измеряет время выполнения каждого алгоритма.

### Вывод результатов: Программа выводит расстояния, полученные каждым из алгоритмов (BFS и DFS), а также время, затраченное на выполнение каждого из алгоритмов.

### Программа позволяет сравнить два подхода к решению задачи поиска кратчайших путей в графе и оценить их эффективность на основе времени выполнения

### **Результаты работы программы**

### 

Рисунок 1 — Результаты работы программы 1.1-2

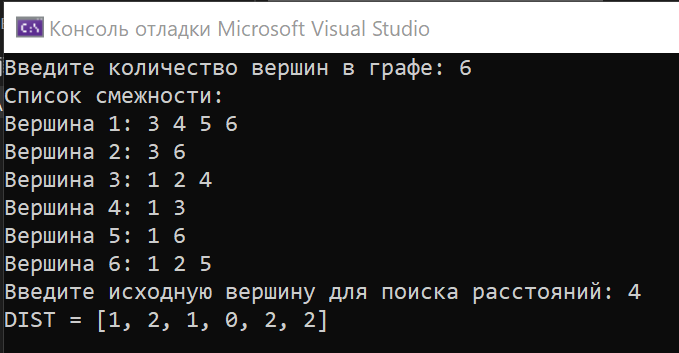


Рисунок 2 — Результаты работы программы 1.3

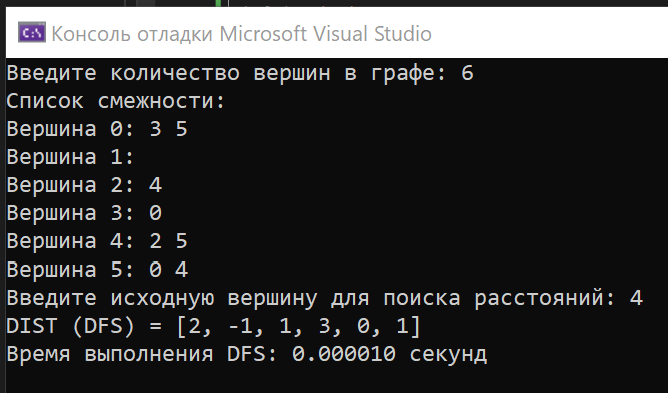


Рисунок 3 — Результаты работы программы 2.1-2

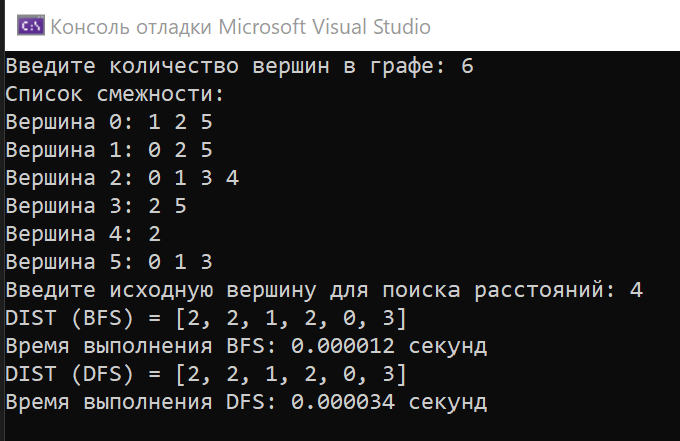


Рисунок 4— Результаты работы программы 2.3

По результатам работы программы, обход в глубину для поиска расстояния оказался более времязатратным.

**Вывод**

Приобретены навыки программирования и работы алгоритмов нахождения расстояния до других вершин графа различными способами.

.