

Санкт-Петербургский Национальный Исследовательский Университет
Информационных Технологий, Механики и Оптики

Факультет инфокоммуникационных технологий и систем связи

Лабораторная работа №5

Вариант №1

Выполнил(и:)

Алексеев Т.Ю.

Оншин Д.Н.

Проверил

Мусаев А.А.

Санкт-Петербург,

2022

СОДЕРЖАНИЕ

ВВЕДЕНИЕ	3
1. ЗАДАНИЕ 2.....	4
2. ЗАДАНИЕ 2.....	6
ЗАКЛЮЧЕНИЕ.....	7
СПИСОК ЛИТЕРАТУРЫ	8

ВВЕДЕНИЕ

Целью данной работы являлось знакомство с алгоритмами поиска в глубину и в ширину в графе.

Для достижения данной цели необходимо было выполнить следующие задания:

1. Задание 2: придумать и решить задания с помощью алгоритмов поиска в глубину и в ширину. Объяснить, почему были выбраны именно эти алгоритмы.
2. Задание 3: написать программу, которая ищет выход из лабиринта, представленного в виде квадратной матрицы

Решения данных задач будут находиться на GitHub по ссылке:
https://github.com/NorthPole0499/Algoritms_task_5

1. ЗАДАНИЕ 2

В данном задании необходимо было придумать два задания для двух алгоритмов: поиска в глубину и поиска в ширину.

Для поиска в глубину балы придумана такая задача «Найти вершину, путь к которой лежит через вершины с номерами 1, 5 и 2». Для её решения была написана рекурсивная функция поиска в глубину, на вход которой подавалось 2 аргумента: номер вершины и пройденный путь, представленный списком. Если номер вершины содержится в ключах словаря и номер вершины совпадает с чем-то из [1, 5, 2], то запускаем рекурсию этой функции для листьев этой вершины. Иначе, если пройденный путь равен [1, 5, 2], то выводим номер вершины, это и есть ответ. Алгоритм поиска в глубину использовался, так как нам нужен именно маршрут от начала графа до его конца, нам не обязательно перебирать все вершины. Куда выгоднее будет выборочно опускаться на нужные уровни графа с нужными вершинами. Этим и обоснован выбор.

Для поиска в ширину была придумана такая задача «Найти уровень дерева, на котором сумма весов наибольшая». Для её решения была написана функция, принимающая на вход только стартовую вершину. Весь код строится на двух списках: first, second. В первый мы записываем вершины текущего уровня, а во второй – вершины, в которые можно попасть из вершин первого списка. После итерации в первый список записывается второй, а второй очищается. Таким образом, программа проходит по каждому уровню и находит сумму всех вершин на каждой итерации, после чего выявляет наибольшую. Поиск в ширину был выбран, так как нам предстояло работать не с маршрутами, а с уровнями графа. Именно из-за этого и был выбран соответствующий алгоритм.

```
Номер вершины  
9  
  
Максимальная сумма 26 находится на уровне номер 4  
  
Process finished with exit code 0
```

Приложение 1 – Пример ввода и вывода программы для Задания 2

2. ЗАДАНИЕ 2

В данном задании нужно написать программу, которая находит кратчайший путь от точки старта до точки финиша в лабиринте.

В начале создается лабиринт (1 – стена, 0 - проход) и указываются старт и финиш, причем лабиринт со всех сторон окружен стенами. Это сделано для удобства, чтобы не выделять частные случаи, когда по краям лабиринта с одной стороны нет стены. Лабиринт представляется в виде матрицы, состоящей из 1 и 0. Далее создается матрица такого же размера, заполненная нулями.

Далее новая матрица заполняется номерами шагов, которые в свою очередь рассчитываются по следующему принципу: если в следующей ячейке нет стены, то берется предыдущий шаг, и к нему прибавляется единица. Это происходит до того момента, пока программа не дойдет до конечной точки. Получается матрица, заполненная номерами шагов.

Далее нужно найти кратчайший путь от старта до финиша. Программа идет от конца до начала и сравнивает значения шагов, если шаг на единицу меньше, чем текущий, значит, это часть пути. Координаты всех точек, лежащих на пути, записываются в список.

После этого в первоначальную матрицу записывается путь по координатам из списка.

ЗАКЛЮЧЕНИЕ

Таким образом, в ходе выполнения данной лабораторной работы были достигнуты все поставленные цели. Произошло знакомство алгоритмами поиска в глубину и в ширину в графе. Также, знания были применены при решении интересной практической задачи с лабиринтом.

СПИСОК ЛИТЕРАТУРЫ

- 1) Фоксфорд: Алгоритм поиска в ширину // URL:
[https://foxford.ru/wiki/informatika/algorithm-poiska-v-shirinu#:~:text=Алгоритм%20поиска%20в%20ширину%20\(англ,путь%20С%20содержащий%20наименьшее%20число%20ребер.](https://foxford.ru/wiki/informatika/algorithm-poiska-v-shirinu#:~:text=Алгоритм%20поиска%20в%20ширину%20(англ,путь%20С%20содержащий%20наименьшее%20число%20ребер.)
(дата обращения: 15.12.2022)