ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ

«НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО»

Отчёт по лабораторной работе № 8

«Алгоритмы нахождения пути в графе»

Выполнил работу

Косарев И. А.

Академическая группа №J3113

Принято

Ходненко И. В.

Санкт-Петербург

2024

**Структура отчёта:**

1. Введение

Цель: познакомиться с различными методами нахождения пути в графе и реализовать их.

Задачи:

* реализовать структуру графа и узлов;
* познакомиться с BFS и реализовать его;
* ознакомиться с DFS и реализовать его;
* ознакомиться с алгоритмом Дейкстры и реализовать его;
* ознакомиться с алгоритмом A\* и реализовать его;
* сравнить их по времени выполнения;
* сделать выводы.

1. Теоретическая подготовка

unordered\_map — это ассоциативный контейнер, который хранит элементы в виде пар "ключ-значение" и обеспечивает быстрое извлечение значений по ключу.

stack — это контейнер, представляющий собой абстрактный тип данных "стек", работающий по принципу LIFO (Last In, First Out — последним пришел, первым ушел).

queue — это контейнер, представляющий собой абстрактный тип данных "очередь", работающий по принципу FIFO (First In, First Out — первым пришел, первым ушел).

priority\_queue — это специальный тип данных, представляющий собой очередь с приоритетом, где каждый элемент имеет приоритет, и элементы извлекаются в порядке убывания их приоритета.

Метод нахождения пути в графе с помощью BFS — это алгоритм поиска в ширину, который исследует вершины графа на одном уровне, прежде чем переходить к вершинам следующего уровня.

Метод нахождения пути в графе с помощью DFS — это алгоритм поиска в глубину, который исследует как можно дальше по каждому пути, прежде чем откатываться назад.

Метод нахождения пути в графе с помощью алгоритма Дейкстры — это алгоритм поиска кратчайшего пути от одной вершины до всех других в графе с неотрицательными весами рёбер. Он использует приоритетную очередь для выбора вершины с минимальным текущим расстоянием.

Метод нахождения пути в графе с помощью алгоритма A — это расширение алгоритма Дейкстры, которое включает в себя эвристическую функцию для оценки стоимости пути от текущей вершины до целевой. Это позволяет алгоритму более эффективно находить кратчайший путь в графах.

1. Реализация

Принцип работы методов и реализация структуры узлов и графа и методов представлены на изображениях ниже.

Изображение выглядит как текст, снимок экрана, дисплей, программное обеспечение

Автоматически созданное описание

Рисунок 1 - структура узла

Изображение выглядит как текст, снимок экрана, программное обеспечение

Автоматически созданное описание

Рисунок 2 - структура графа + доп. функции (нужны ниже)

Изображение выглядит как текст, снимок экрана

Автоматически созданное описание

Рисунок 3 - функции поиска узла по координатам и функция записи из файла

Изображение выглядит как текст, снимок экрана, программное обеспечение, дисплей

Автоматически созданное описание

Рисунок 4 - функция нахождения приблизительного узла и функтор для приоритетной очереди

Изображение выглядит как текст, снимок экрана

Автоматически созданное описание

Рисунок 5 - метод нахождения пути на основе BFS

Изображение выглядит как текст, снимок экрана, дисплей, программное обеспечение

Автоматически созданное описание

Рисунок 6 - метод нахождения пути на основе DFS

Изображение выглядит как текст, снимок экрана

Автоматически созданное описание

Рисунок 7 - метод нахождения кратчайшего пути на основе алгоритма Дейкстры

Изображение выглядит как текст, снимок экрана, дисплей, программное обеспечение

Автоматически созданное описание

Рисунок 8 - метод нахождения кратчайшего пути на основе алгоритма A\*

1. Экспериментальная часть

Время выполнения на паре узлов представлено на изображении ниже.

Изображение выглядит как текст, Шрифт, снимок экрана

Автоматически созданное описание

Рисунок 9 - время выполнения для каждого метода

Все алгоритмы работают за сравнительно малое время, так как имеют приблизительно линейную сложность (каждое ребро / узел рассматривается лишь один раз).

Причиной, почему A\* может работать немного дольше, чем Дейкстра, может являться то, что в данном примере кол-во узлов в кратчайшем пути достаточно невелико, из-за чего перезапись в массиве предположительных расстояний и расчет эвристики значительно замедлили выполнение алгоритма.

1. Заключение

В процессе выполнения лабораторной работы я познакомился с различными методами нахождения пути в графе, а также реализовал их. Цель работы была достигнута посредством тестирования данных реализованных методов на различных графах. Был проведен анализ времени выполнения алгоритмов.  
В качестве дальнейших исследований можно предложить тестирование на большем кол-ве различных наборах входных данных (графах).