



UNIOESTE

Universidade Estadual
do Oeste do Paraná

Projeto 1 - Algoritmos de Busca

Professora:

Huei Diana Lee

Alunos:

Victor Hugo Almeida Alicino,
Victor Emanuel Almeida

December 4, 2022

Conteúdo

1	Introdução	2
2	Executando o programa	2
2.1	Requisitos	2
2.2	Compilando	2
2.3	Executando	3
3	Implementação	3
3.1	Estruturas de dados	3
3.2	Algoritmos de busca	3
3.2.1	Algoritmo A* (melhor solução)	3
3.2.2	Busca em largura (pior solução)	3
3.2.3	Busca em profundidade (extra)	3
4	Resultados	3
4.1	Descrição do teste	3
4.2	Resultados dos testes	4
5	Conclusão	4

1 Introdução

O Projeto tem como objetivo a implementação de um sistema para auxiliar turistas em Veneza a chegarem aos diversos museus da cidade.

Implementado inteiramente em `c++20` sem utilização de bibliotecas externas além das bibliotecas padrão do `c++20`, podendo assim ser compilado em qualquer sistema operacional que possua o compilador adequado.

Essa linguagem foi escolhida pois além de possuir abstrações alto nível com o uso de classes, ainda é extremamente eficiente ao ser compilada diretamente para linguagem de máquina. Além do disso a dupla responsável pelo projeto possui familiaridade com a linguagem.

O sistema é capaz de calcular a rota mais curta entre dois pontos, utilizando os algoritmos A^* , busca em largura e busca em profundidade.

Visando a simplicidade e levar os algoritmos a seus limites de eficiência, o sistema não implementa uma interface gráfica, todos os argumentos devem ser passados como argumentos de linha de comando, ou coletados em tempo de execução, para ver os argumentos para o programa veja a seção 2.3.

Uma vez que todos os parâmetros para execução do programa podem ser passados antes que ele inicie, pode-se fazer uso de scripts para automatizar a execução do programa gerando resultados para diferentes cenários. Dessa maneira foi implementado um script para testar o desempenho, explicado na seção 4.

2 Executando o programa

2.1 Requisitos

- `cmake`;
- `make`;
- `g++`;
- `git` (opcional).

2.2 Compilando

- `cd build`
- `cmake .`
- `make`

2.3 Executando

Para executar o programa basta executar o arquivo

`Graph_Search_Algorithms_1.0.0` gerado na pasta `build`.

Caso nada seja passado como argumento, o programa irá coletar os argumentos em tempo de execução, sendo eles:

1. O caminho para o arquivo de entrada;
2. O caminho para o arquivo de saída contendo o resumo da execução;
3. O algoritmo a ser utilizado, podendo ser:
 - $A \rightarrow A^*$;
 - $B \rightarrow BFS$, *Breadth First Search* (busca em largura);
 - $D \rightarrow DFS$, *Depth First Search* (busca em profundidade).

3 Implementação

3.1 Estruturas de dados

3.2 Algoritmos de busca

3.2.1 Algoritmo A^* (melhor solução)

3.2.2 Busca em largura (pior solução)

3.2.3 Busca em profundidade (extra)

4 Resultados

Com o sistema implementado, foi possível realizar os seguintes testes:

4.1 Descrição do teste

- Para cada um dos algoritmos de busca implementados, foi executado X vezes passando os parâmetros *arquivo_saida_i* *algoritmos_j*, sendo $i \in \{1, \dots, X\}$ e $j \in \{A^*, \dots\}$.
- Os testes foram realizados em um computador com as seguintes características:
 - **Processador:** i3-1115G4 4.100GHz;

- **Memória RAM:** 8GB;
 - **SSD:** 256GB;
 - **Sistema operacional:** Arch Linux ⁱ.
- A execução dos testes foi realizada logo após a inicialização do computador, sem nenhum outro processo de usuário em execução.

4.2 Resultados dos testes

5 Conclusão

ⁱLink para download do sistema operacional <<https://archlinux.org/download/>>