

## **Trabalho Prático – Busca de Rotas em Grafos**

**Unidade Curricular:** Estrutura de Dados e Análise de Algoritmos – 2º / 2025

**Professor:** Lucas Goulart Silva

**Prazo de entrega:** 28/11/2025

**Valor:** 25 pontos

**Objetivo:** Este trabalho tem como objetivo implementar e comparar algoritmos clássicos de busca de rota em grafos, aplicados a grades bidimensionais (matrizes) com pesos e obstáculos.

### **Tarefas:**

#### **1. Implementar os algoritmos**

- a. Busca em largura (**Breadth-First-Search - BFS**);
- b. Busca em profundidade (**Deep-First-Search - DFS**);
- c. **Dijkstra**;
- d. **Ganancioso** - Greedy Best-First-Search - com heurísticas Euclidiana e Manhattan;
- e. **A\*** - com heurísticas Euclidiana e Manhattan.

#### **2. Avaliar os algoritmos em diferentes cenários, medindo:**

- Caminho encontrado e o seu custo;
- Número de nós visitados / expandidos;
- Tempo de execução.

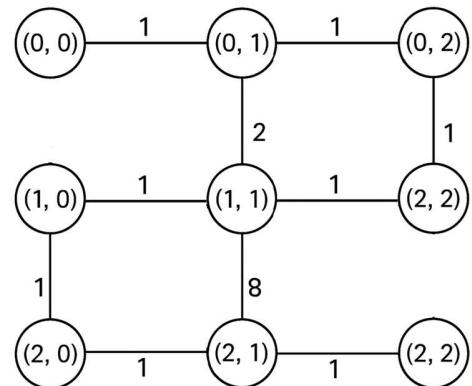
**Descrição do Problema:** O aluno deve desenvolver um programa capaz de ler uma matriz representando um grafo bidimensional com pesos e obstáculos. Dados os nós de origem e destino (como parâmetros de execução), executar cada um dos 5 algoritmos para encontrar uma rota. Quando aplicável, utilizar as duas heurísticas possíveis. Gerar arquivos de saída padronizados contendo resultados e métricas de desempenho.

**Arquivo de Entrada:** Um arquivo de entrada contém uma matriz adjacências, representando o grafo usando uma tabela bidimensional. Nessa matriz de adjacências cada linha representa um nó de origem do grafo, e cada coluna

representa um nó de destino. O valor que aparece na interseção de uma linha e uma coluna indica o custo ou peso para ir do nó de origem até aquele nó de destino. Se o valor menor ou igual a zero significa que não há conexão direta entre os dois nós. Dessa forma, é possível olhar qualquer linha da matriz e entender quanto custa, ou se é possível, acessar todos os outros nós a partir daquele nó de origem.

Observe, como exemplo a matriz a seguir, e o grafo que a mesma representa:

0	1	0	0	0	0	0	0	0
1	0	1	0	2	0	0	0	0
0	1	0	0	0	1	0	0	0
0	0	0	0	1	0	1	0	0
0	2	0	1	0	1	0	8	0
0	0	1	0	1	0	0	0	0
0	0	0	1	0	0	0	1	0
0	0	0	0	8	0	1	0	1
0	0	0	0	0	0	0	1	0



**Execução do Programa:** O programa deve ser executado informando o Arquivo de entrada, o vértice de origem e o vértice de destino. Observe o formato do comando:

```
java MeuPrograma teste_3x3.txt 0 8
```

**Saídas do Programa:** Para cada execução, o programa deve gerar 6 arquivos de saída padronizados:

- teste\_3x3.txt.dfs
- teste\_3x3.txt.bfs
- teste\_3x3.txt.dijkstra
- teste\_3x3.txt.a.manhattan
- teste\_3x3.txt.a.euclidiana
- teste\_3x3.txt.gbs.manhattan
- teste\_3x3.txt.gbs.euclidiana

O formato de saída esperado para, por exemplo, o algoritmo A\*, utilizando a heurística Manhatann é:

ALGORITMO: A\*

HEURISTICA: Manhattan

ORIGEM: 0

DESTINO: 8

CAMINHO: 0 -> 1 -> 4 -> 3 -> 6 -> 7 -> 8

CUSTO: 7

NOS EXPANDIDOS: 9

TEMPO (ms): 2.43

- Caso um caminho válido não tenha disso encontrado, o seu valor na saída deve ser deixado em branco.
- Caso uma heurística não tenha sido utilizada, o seu valor na saída deve ser deixado em branco.

#### Regras e Convenções:

- Custo total: soma dos pesos das arestas percorridas, a partir do primeiro movimento;
- Bidirecionalidade: se um vértice A é acessível a partir de outro vértice B, o caminho de volta também é permitido.

#### Algoritmos e Heurísticas:

ALGORITMO	HEURÍSTICAS
BFS	-
DFS	-
DIJKSTRA	-
GREEDY BEST-FIRST-SEARCH	Manhattan, Euclidiana
A*	Manhattan, Euclidiana

**Entregas:**

1. **Código-fonte** completo (Java ou C#);
2. **Arquivos de saída** gerados para cada teste;
3. **Relatório** comentando os resultados observados nos testes, incluindo:
  - a. Para os casos em que há heurística, ela foi determinante para os resultados?
  - b. Algun dos algoritmos apresentou melhor performance? Se sim, em quais casos?
  - c. O tamanho do grafo testado impacta a performance dos algoritmos? De que forma?

Utilize recursos como **gráficos** e **tabelas** para embasar as respostas e justificar as observações.

Atenção: não é necessário se limitar às perguntas acima. Quaisquer padrões identificados nos testes podem ser considerados resultados válidos, desde que sejam justificados pelos dados obtidos.