

# T2 - Implementação de algoritmo utilizando programação dinâmica

## Análise de Algoritmo

Última ocorrência: 2023/02 em Universidade Lasalle Canoas

---

Implementar o algoritmo para encontrar o caminho entre dois pontos em um mapa, levando em consideração o custo de locomoção nesse mapa. O mapa é um grid definido como uma série de valores inteiros. Se o valor for positivo ou igual a zero, representa o custo (esforço) de passar pelo terreno. Se o valor for negativo (-1), significa que é impossível passar pelo terreno. Crie uma interface e representação da resposta interessante para a utilização do sistema implementado no item 1.

## Objetivo

Praticar a implementação de um algoritmo utilizando programação dinâmica.

## Tarefas

Implementar um programa que leia um mapa contendo a descrição de um ambiente e a posição inicial de um objeto e crie um caminho entre esse objeto e uma posição arbitrária, escolhida pelo usuário do sistema.

O programa deverá executar as seguintes operações:

- Ler um mapa em arquivo texto contendo a descrição do ambiente e a posição inicial do objeto que deve ser movido. O formato do mapa é como segue:

```
<width:int> <height:int>
<x:int> <y:int>
<múltiplas linhas com valores inteiros a partir de -1>
```

- Exemplo de mapa

```
10 8
0 7
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 -1 -1 -1 -1 -1 0 0
0 0 0 0 0 0 0 -1 0 0
0 0 0 0 0 0 0 -1 0 0
0 0 0 0 0 0 0 -1 0 0
0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0
```

- Os valores em cada célula do mapa representam o **custo extra** de passar por aquele ponto. O valor **-1** deve ser tratado como um obstáculo intransponível (com custo **infinito**).

- 

As coordenadas do mapa começam em **(0,0)** que representa o canto *superior esquerdo*.

No exemplo, o objeto está na coordenada **(0,7)**, que no mapa representado representa o canto inferior esquerdo.

- 

Após ler o mapa, esperar que o usuário forneça as coordenadas para onde o objeto deve ser movido.

-

Calcular a melhor rota (menor custo) entre o ponto original do objeto e as coordenadas providas pelo usuário, utilizando o **algoritmo A\***

- 

Cada movimento tem custo mínimo de **1**, e o objeto só pode se mover na horizontal (eixo **x**) e na vertical (eixo **y**). Não há custo extra para trocar o eixo de movimento.

- Como saída, o programa deve listar o caminho escolhido contendo o custo total do trajeto e as coordenadas a serem percorridas no seguinte formato:

```
<custo:int> <x:int>,<y:int> <x:int>,<y:int>...
```

- Por exemplo, saindo da posição **(1, 1)** e terminando na posição **(6, 4)**, com um custo de 123, a saída seria:

```
123 1,1 2,1 2,2 2,3 2,4 3,4 4,4 5,4 6,4
```

- Exemplo de mapa com custo de movimento variável:

```
10 8
0 7
3 1 4 -1 -1 8 8 8 0 0
3 3 2 6 2 3 8 8 1 0
3 3 3 5 6 8 10 6 1 0
0 1 1 4 0 10 10 -1 1 0
0 1 1 4 5 0 0 -1 1 0
0 1 1 1 1 0 0 -1 1 0
0 1 1 1 1 1 10 10 3 0
0 0 0 0 0 0 10 10 10 0
```

## Artefatos

Deve ser entregue apenas o *link* para um repositório público no [Github](#) ou outro serviço de armazenamento de repositórios [Git](#).

O repositório deverá conter a implementação do trabalho, e um arquivo README (sugere-se o formato Markdown e um arquivo README.md) contendo instruções para a execução do sistema.

Data de Entrega: 2023-12-04

## Observações

- O trabalho pode ser realizado em grupos de até 3 pessoas.
- Os testes do programa serão executados em mapas pré-definidos no formato apresentado.
- Em caso de plágio, a nota atribuída ao trabalho será 0 (zero).

## Recursos para a elaboração deste trabalho

- [Notas de aula sobre Programação Dinâmica](#)
- [Algoritmo A\\*](#) Wikipedia (inglês)
- [Algoritmo A\\*](#) Geeks For Geeks (inglês)