

Universidade Federal de Ouro Preto  
PCC104 - Projeto e Análise de Algoritmos  
Força Bruta e Busca Exaustiva

Prof. Rodrigo Silva

September 21, 2024

## Instruções

### 1 Leitura Recomendada

- Capítulo 3 - *Introduction to the Design and Analysis of Algorithms (3rd Edition)* - Anany Levitin
- Livro - *Problem Solving with Algorithms and Data Structures using C++* (disponível em: <https://runestone.academy/runestone/books/published/cppds/index.html#>)
- Arrays <https://www.interviewcake.com/concept/python/array?>
- LinkedLists <https://www.interviewcake.com/concept/python/linked-list?>
- Pilhas <https://www.interviewcake.com/concept/python/stack?>
- Filas <https://www.interviewcake.com/concept/python/queue?>
- Livro - *Introdução à programação* - Alan de Freitas (disponível em <http://www.decom.ufop.br/alan/bcc702/livrocpp.pdf>)

### 2 Atividades Práticas

1. Implementar o algoritmo *Selection Sort*
2. Implementar o algoritmo *SequentialSearch2* (Ver Seção 3.2 *Introduction to the Design and Analysis of Algorithms (3rd Edition)* - Anany Levitin).
3. Implemente uma solução baseada em busca exaustiva para o problema do Caixeiro Viajante (Traveling Salesman Problem).
4. Implemente uma solução baseada em busca exaustiva para o problema da Mochila binário (Binary Knapsack Problem).
5. Dada uma grade de tamanho  $n \times n$  preenchida com 0, 1, 2, 3. Verifique se há um caminho possível do ponto de origem ao destino. Você pode percorrer para cima, para baixo, para a direita e para a esquerda.

#### Descrição das Células:

- Um valor de célula 1 significa Origem.
- Um valor de célula 2 significa Destino.
- Um valor de célula 3 significa célula vazia.
- Um valor de célula 0 significa Parede (célula bloqueada que não pode ser atravessada).

**Nota:** Há apenas uma única origem e um único destino.

**Exemplos:**

• **Entrada:**  $\text{grid} = \begin{bmatrix} 3 & 0 & 3 & 0 & 0 \\ 3 & 0 & 0 & 0 & 3 \\ 3 & 3 & 3 & 3 & 3 \\ 0 & 2 & 3 & 0 & 0 \\ 3 & 0 & 0 & 1 & 3 \end{bmatrix}$

**Saída:** 0

**Explicação:** A grade é como abaixo:

$$\begin{bmatrix} 3 & 0 & 3 & 0 & 0 \\ 3 & 0 & 0 & 0 & 3 \\ 3 & 3 & 3 & 3 & 3 \\ 0 & 2 & 3 & 0 & 0 \\ 3 & 0 & 0 & 1 & 3 \end{bmatrix}$$

Não há caminho para chegar ao destino (3,1) a partir da origem (4,3).

• **Entrada:**  $\text{grid} = \begin{bmatrix} 1 & 3 \\ 3 & 2 \end{bmatrix}$

**Saída:** 1

**Explicação:** A grade é como abaixo:

$$\begin{bmatrix} 1 & 3 \\ 3 & 2 \end{bmatrix}$$

Há um caminho da origem (0,0) ao destino (1,1).

**Complexidade de Tempo Esperada:**  $O(n^2)$

**Complexidade de Espaço Auxiliar Esperada:**  $O(n^2)$

Para cada implementação, apresentar a análise de complexidade de pior caso e melhor caso (se houver) do tempo do algoritmo. Esta análise deverá conter:

- Expressão matemática que define o número de operações (relação de recorrência para recursivos ou somatórios para iterativos)
- Expressão final da função de custo
- Indicação da classe de eficiência ( $O$  ou  $\Theta$ ). A indicação da classe, deve ser justificada. Você pode provar pela definição, pelo limite, ou utilizar resultado demonstrados na primeira lista de exercícios (referentes ao capítulo 2 do livro).