

Universidade Federal de Ouro Preto
PCC104 - Projeto e Análise de Algoritmos
Prova 2

Prof. Rodrigo Silva

October 17, 2023

Orientações

- É obrigatória a entrega do código fonte dos algoritmos implementados. Provas sem os códigos fonte não serão corrigidas e terão nota 0.
- O código não deve conter nenhuma informação/comentário que auxilie na análise de complexidade do mesmo.
- A avaliação do código apresentado entra na avaliação das questões relacionadas.
- Simplificações feitas na análise de custo dos algoritmos devem ser indicadas e justificadas.

Questões

1. Considere a sua implementação do algoritmo de Busca Binária
 - (a) (1.5 pt) Apresente a expressão matemática que define o custo em termos do número de comparações. Explique o que representa e de onde saiu cada um dos termos da sua expressão.
 - (b) (1 pt) Resolva a expressão para encontrar o custo do algoritmo.
 - (c) (1 pt) Determine a classe de complexidade em notação O ou Θ utilizando algum método formal.
2. Considere a sua implementação do algoritmo que resolve o problema da moeda falsa.
 - (a) (1.5 pt) Apresente a expressão matemática que define o custo em termos do número de operações de adição. Explique o que representa e de onde saiu cada um dos termos da sua expressão.
 - (b) (1 pt) Resolva a expressão para encontrar o custo do algoritmo.
 - (c) (1 pt) Determine a classe de complexidade em notação O ou Θ utilizando algum método formal.
3. Considere a implementação de uma árvore de busca binária abaixo.

```
1 class BST:
2     def __init__(self):
3         self.root = None
4
5     def search(self, value):
6         return self._search(self.root, value)
7
8     def _search(self, node, value):
```

```

9         if node is None:
10             return False
11         elif node.value == value:
12             return True
13         elif value > node.value:
14             return self._search(node.right, value)
15         elif value < node.value:
16             return self._search(node.left, value)
17
18 class Node:
19     def __init__(self, value):
20         self.value = value
21         self.left = None
22         self.right = None

```

- (a) (1 pt) Apresente a expressão matemática que define o custo do algoritmo de busca (**search**) no **pior caso** em termos do número comparações.
- (b) (1 pt) Apresente a expressão matemática que define o custo do algoritmo de busca (**search**) no **melhor caso** termos do número comparações.
- (c) (1 pt) O melhor caso do algoritmo de busca em uma árvore de busca binária é $\Theta(\log_2 n)$. Discuta vantagens e desvantagens desta estrutura de dados em relação a busca em um vetor utilizando busca binária ou não.