

ANÁLISE E PROJETO DE ALGORITMOS – REAVALIAÇÃO

Prof. Thiago Cavalcante

- Não use celular/computador e não converse com ninguém, a prova é individual.
- Sinta-se à vontade para tirar dúvidas (**razoáveis**) ou pedir esclarecimentos sobre as questões.
- Use **letra legível!** não posso dar nota para algo que não consigo ler.
- Lembre-se de **assinar seu nome nas suas folhas**. Se usar **mais de uma** folha, **enumere cada página**.
- **Seja organizado:** especifique número e letra da questão que você está respondendo e deixe um espaço entre as respostas, para não ficar tudo amontoado. Você pode pegar mais folhas, se precisar.

NOME: _____

1. (3,0 pt) Use o princípio da indução para provar as afirmações a seguir.

- (a) (0,8 pt) $6^n + 4$ é divisível por 5, para todo $n \geq 0$.
- (b) (1,0 pt) $5^{2n+1} + 2^{2n+1}$ é divisível por 7, para todo $n \geq 0$.
- (c) (1,2 pt)

$$\sum_{i=1}^n \frac{1}{i(i+1)} = \frac{n}{n+1},$$

para todo $n \geq 1$.

2. (2,0 pt) Considere o pseudocódigo abaixo:

Para i de 0 até n - 1:
 Para j de i + 1 até n - 1:
 Comando

- (a) (0,5 pt) Expresse como um somatório a quantidade de vezes que a linha Comando é executada.
- (b) (1,0 pt) Simplifique o somatório até chegar a uma fórmula em função de n. Para isso, use a expressão abaixo:

$$\sum_{i=0}^{n-1} i = \frac{n(n-1)}{2}$$

- (c) (0,5 pt) De acordo a fórmula, qual é o tempo de execução de pior caso em notação Big-Oh?

3. (1,5 pt) Ordene as funções a seguir por sua dominância, da mais dominante para a menos dominante:

- n^3
- n
- \sqrt{n}
- $n \log n$
- $\log n$
- n^2
- $n!$
- c^n
- 1

4. (1,5 pt) Mostre, usando as definições de $O(f(n))$ e $\Omega(f(n))$ que:

- (a) Se $f(n) = O(g(n))$ e $g(n) = O(h(n))$, então $f(n) = O(h(n))$
- (b) Se $f(n) = \Omega(g(n))$, então $g(n) = O(f(n))$

5. (1,0 pt) Relacione cada complexidade de tempo com uma operação em um algoritmo.

- (a) $O(n^2)$
- (b) $O(1)$
- (c) $O(n!)$
- (d) $O(\log n)$
- (e) $O(n)$
- ☐ Percorrer um array do início ao fim
- ☐ Extrair de um array um elemento de índice x
- ☐ Percorrer uma matriz do início ao fim
- ☐ Gerar todas as permutações de um conjunto de dados
- ☐ Fazer uma busca binária em um array ordenado

6. (1,0 pt) Para cada par de funções, especifique se $f(n) = O(g(n))$ ou $f(n) = \Omega(g(n))$. Lembre-se das relações de dominância.

- (a) $f(n) = \sqrt{n}$, $g(n) = \log(n^2)$
- (b) $f(n) = 6n^2$, $g(n) = n^2 \log n$
- (c) $f(n) = n \log n + n$, $g(n) = \log n$
- (d) $f(n) = 10n^2$, $g(n) = 2^n$
- (e) $f(n) = (\log n)^2$, $g(n) = \log n$