

Nome:
RG:

QUESTÕES

Questão 1. (peso 1) Um programador tenta relembrar uma sequência de 11 bits que dá acesso ao seu cofre digital. No entanto, ele se recorda apenas dos três primeiros e dos dois últimos bits, que são 110 e 00, respectivamente. Sabendo apenas disso, qual a probabilidade do programador acertar um palpite de sequência?

Questão 2. (peso 1,5) Uma cidade possui 100.000 moradores e 3 jornais: A, B e C. A proporção de moradores que leem esses jornais são como seguem:

A: 10% A e B: 8% A e B e C: 1%
B: 30% A e C: 2%
C: 5% B e C: 4%

Existem, por exemplo, 8.000 pessoas que leem os jornais A e B. Responda as questões abaixo:

1. Qual o número de pessoas que leem apenas um jornal?
2. Quantas pessoas leem pelo menos dois jornais?
3. Se os jornais A e C são de bairro e B da mesorregião da cidade, quantas pessoas leem pelo menos um dos jornais locais e o jornal da mesorregião?
4. Quantas pessoas não leem nenhum jornal?
5. Quantas pessoas leem apenas um jornal local e o jornal da mesorregião?

Questão 3. (peso 0,75) Considere a função $f: \mathbb{R} \rightarrow \mathbb{R}$ definida da seguinte forma:

$$f(x) = \begin{cases} x^2 & \text{se } x \leq 0, \\ x^2 + 1 & \text{se } x > 0. \end{cases}$$

Assinale a alternativa correta:

- (a) $\lim_{x \rightarrow 0^-} f'(x) = \lim_{x \rightarrow 0^+} f'(x)$ mas $f'(0)$ não existe.
- (b) $\lim_{x \rightarrow 0^-} f'(x) = 0$ e $\lim_{x \rightarrow 0^+} f'(x) = 1 = f(0)$.
- (c) $f(x)$ é contínua mas não é diferenciável.

(d) $f'(x)$ é decrescente e $f(x) \geq 0$ se $x \in (-\infty, 0)$.

(e) $\lim_{x \rightarrow \infty} f(x) = \infty$ e $\lim_{x \rightarrow -\infty} f'(x) = +\infty$.

Questão 4. (peso 0,75) Determine qual das seguintes proposições **não** pode ser provada a partir da premissa $((a \wedge b) \vee c) \wedge (c \rightarrow d)$:

(a) $(a \vee d) \wedge (b \vee d)$

(b) $\neg a \vee \neg b \rightarrow (c \wedge d)$

(c) $(a \wedge b) \rightarrow \neg d$

(d) $\neg a \rightarrow d$

(e) $\neg d \rightarrow b$

Questão 5. (peso 1) Qual o valor devolvido pela função a seguir quando ela recebe $n = 27$? Explique sua resposta listando todas as chamadas recursivas que são feitas.

```
1 function magic(int n) {
2     if (n <= 10)
3         return 2*n;
4     else
5         return magic(magic(n/3));
6 }
```

Questão 6. (peso 1) Para o conjunto $\{1, 4, 5, 10, 16, 17, 21\}$ de chaves, desenhe árvores binárias de busca com alturas 2, 3, 4, 5 e 6.

Questão 7. (peso 1,5) Apresente, em pseudocódigo, como implementar uma fila utilizando duas pilhas. Explique sua ideia. Analise o tempo de execução das operações de fila em função do número n de elementos armazenados.

Questão 8. (peso 1) Seja T uma árvore AVL vazia. Suponha que os elementos 37, 68, 81, 42, 55, 74, 70 e 10 sejam inseridos, nessa ordem, em T . Qual das sequências a seguir corresponde a um percurso pós-ordem em T ?

(a) 10, 37, 55, 42, 70, 81, 74, 68.

(b) 10, 37, 42, 55, 68, 70, 74, 81.

(c) 81, 74, 70, 68, 55, 42, 37, 10.

(d) 81, 70, 74, 55, 10, 37, 42, 68.

(e) 68, 42, 37, 10, 55, 74, 70, 81.

(f) 68, 74, 81, 70, 42, 55, 37, 10.

Questão 9. (peso 1,5) Seja L uma lista ligada cujas chaves são números inteiros. Responda os itens a seguir.

- (a) Descreva algoritmos eficientes (que minimizem o número de comparações necessárias) em pseudocódigo ou na linguagem imperativa de sua preferência para os seguintes procedimentos:
- **remove_min**(L): função que recebe uma lista L e remove o primeiro nó encontrado que contenha a chave mínima, devolvendo a nova lista L .
 - **verifica_ordem**(L): função que recebe uma lista L e verifica se as chaves em L estão em ordem não-decrescente, devolvendo *Verdadeiro* caso estejam e *Falso* caso contrário.
 - **remove_repeticoes**(L): função que recebe uma lista L e remove nós com chaves duplicadas, devolvendo a nova lista L . Atenção: se dois ou mais nós têm chave x , então ao final deste procedimento a lista deve ter um único nó com chave x .
- (b) Determine o número de comparações necessárias para realizar cada um dos procedimentos acima, no pior caso, em função da quantidade n de elementos armazenados na lista L .
- (c) Realize um teste de mesa (simulação passo a passo) para cada um dos algoritmos descritos considerando $L = [34, 75, 13, 42, 75, 16, 21, 13]$.