

PCC104 – Prova 2 (Versão B)

Universidade Federal de Ouro Preto

1. (2pts) Resolva a seguinte recorrência assumindo que n é potência de 2:

$$T(n) = T\left(\frac{n}{2}\right) + n, \quad T(1) = 1$$

2. (1pt) Apresente um algoritmo para o qual a recorrência da Questão 1 represente o número de operações realizadas no pior caso. Justifique sua escolha.
3. (1pt) Considere a função `sum_digits(n)`, definida de forma recursiva para retornar a soma dos dígitos de um número inteiro positivo n .

Mostre a árvore de chamadas da função ao executar `sum_digits(742)`.

4. (2pts) Considere sua implementação da função `is_palindrome(s)`, que verifica se uma string é palíndroma de forma recursiva. Apresente a análise de complexidade no pior caso.
5. (4pts) *Considere um vetor V de números inteiros distintos e com tamanho n , onde n é potência de 2. Um número x é chamado de *dominante local* se $x > x_{\text{vizinho à esquerda}}$ e $x > x_{\text{vizinho à direita}}$.

Escreva um algoritmo `find_local_dominant(V)` baseado em divisão e conquista que retorna o índice de um elemento dominante local (caso exista).

- Justifique a complexidade da sua solução.
- **Dica:** Avalie o elemento central de V . Caso ele não seja dominante local, em qual lado (esquerdo ou direito) você deveria continuar buscando? Use os vizinhos para decidir.

6. (4pts) *Uma string contém letras minúsculas e o caractere '#', que representa um backspace (remove o último caractere inserido).

Exemplo: `"ab#c"` \rightarrow `"ac"`

Implemente um algoritmo `processa(s)` que retorna a string final após aplicar os backspaces. Use uma pilha.

Justifique a complexidade da sua solução.

- * Você deve escolher ou a questão 5 ou a questão 6.