MINI-PROVA 2025.1
CIN0009 - Algoritmos e Estruturas de Dados

Questão 1:

Considere o vetor **V = (5, 4, 1, 0, 6, 2, 3, 7).**

a) Ilustre a execução do algoritmo **Quick Sort** sobre este vetor, exibindo o seu conteúdo imediatamente após cada execução da função partition. Considere que o pivô escolhido é sempre o **primeiro** elemento do trecho a ser particionado.

b) Agora, ilustre a execução do algoritmo **Merge Sort** sobre o mesmo vetor original V. Exiba o estado do vetor após cada operação de *merge* (intercalação) ser finalizada.

c) **(EXTRA)** Considerando o vetor **V** finalmente ordenado, indique quais os elementos de **V** seriam comparados com o valor x = 4 durante uma **busca binária**, e em qual ordem essas comparações ocorreriam.

Questão 2:

Compare as funções sentenças abaixo e indique se são verdadeiras (V) ou falsas (F).

1. [ ]Tendo $f(x) = x^4 - 4x + 5$ e $g(x) = 4x + 100$, temos $x^2 = O(f(x))$ e $x^3 = \Omega\ (g(x))$.
2. [ ]Tendo $f(x) = x + 156$ e $g(x) = x^5 + 3$, temos $x/100 = O(f(x))$ e $2^\wedge x = \Omega\ (g(x))$.
3. [ ]Tendo $f(x) = x^8 + 8$ e $g(x) = 3x^8$, temos $x^8 = O(f(x))$ e $x^8 = \Omega\ (g(x))$.
4. [ ]Tendo $f(x) = x^{100} - 2x - 3$ e $g(x) = x^{21} + 300$, temos $x^{99} = O(f(x))$ e $x^{16} + 750= \Omega\ (g(x))$.
5. [ ]Tendo $f(x) = x^{37} + 28x - 20$ e $g(x) = x^{25} - 3x$, temos $x = O(f(x))$ e $x = \Omega\ (g(x))$.

Questão 3:

Dada a seguinte sequência de inserção de uma árvore AVL : 70, 50, 90, 80, 85, 60, 65, 40, 30, 35. Mostre como exatamente a árvore ficará após a inserção do número 80 e do 35.

Questão 4:

Após aprender sobre estruturas como vetor, **pilha**, **fila** e **lista duplamente encadeada**, um habitante de Marte desenvolveu um programa de computador com o seguinte comportamento:

- Se o resto da divisão do inteiro de entrada por 2 for igual a 0, ele será adicionado a uma pilha.
- Se o resto da divisão do inteiro de entrada por 2 for diferente de 0, ele será adicionado a uma fila.

Após o término das entradas, os números são selecionados de forma alternada da fila e da pilha para serem adicionados a uma lista duplamente encadeada (o elemento será adicionado no fim da lista) até que ambas as estruturas (fila e pilha) estejam vazias.

Os elementos da lista duplamente encadeada são, então, adicionados a um vetor a partir da sua primeira posição (vetor[0]), mas em ordem inversa: o último número adicionado na lista será o primeiro do vetor, e o primeiro a ser adicionado na lista será o último do vetor. Por fim, um heap máximo é feito nesse vetor.

A. Considerando a entrada [1, 2, 3, 4, 5, 6, 7, 8, 9, 10] mostre como **fica a pilha e a fila.**
B. Considerando as estruturas de pilha e fila da questão anterior mostre como ficou a lista e o vetor após a inserção dos elementos.
C. Faça o heap máximo e mostre o estado final do vetor.

Questão 5:

Dados um vetor de inteiros V = (v_0, …, v_{n-1}) e um inteiro S, queremos encontrar dois elementos v_i e v_j de V tais que v_i+v_j = S. Esse problema pode ser resolvido com auxílio de uma tabela de dispersão (**Hashtable**) da seguinte maneira.

Inicie com uma hashtable H vazia. Para cada i = 0,…,n−1, procure d = S − v_i em H. Caso encontre, retorne (d,v_i). Caso não encontre, insira v_i em H e continue. Se, após haver inserido todos os elementos de V, ainda não tiver encontrado um par de soma S, é porque tal par não existe, e nesse caso retorne $\perp$.

Considere uma hashtable de **endereçamento fechado** com tamanho inicial m = 3 com função de dispersão **h(k) = k(mod m)**. Nesta tabela, imediatamente antes de cada inserção, se o fator de carga for maior que o limite α_{max}=0.5, executa-se o rehashing, atualizando o tamanho da tabela para 2m+1.

Ilustre a execução do algoritmo sobre a entrada

V = (28, 54, 25, 60, 14, 44, 45) e S = 70,

exibindo a tabela após cada inserção realizada. Indique o que o algoritmo retorna neste caso.