



EXERCÍCIO - Aula 07 – HEAP

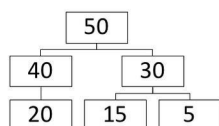
Pedro Vinícius da Silva Ribeiro Mat: 2019033903

1) Enuncie a(s) característica(s) que define(m) um heap e diga se cada um dos grafos a seguir são ou não são um heap.

R = Um heap é uma árvore binária completa em que todos os filhos de um determinado nó são menores (min-heap) ou maiores (max-heap) que ele e no último nível as folhas estão mais à esquerda possível

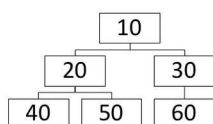
A) Não é Heap

a)



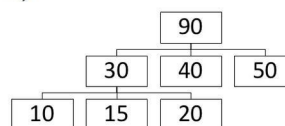
B) Min-Heap

b)

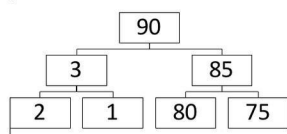


C) Não é Heap (nem binária)

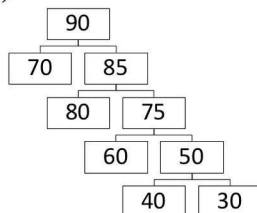
c)



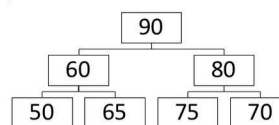
d)



e)



f)



D) Max-heap

E) Não é Heap

F) Não é Heap

2) Considere o vetor de heap = (95,60,78,39,28,66,70,33). Mostre passo a passo quais seriam as modificações que o vetor sofre quando:

a. é retirado o elemento do topo;

1. Vetor [95, 60, 78, 39, 28, 66, 70, 33];
2. [33, 60, 78, 39, 28, 66, 70] heap[0] = heap.pop(-1);
3. Entra na função recursiva shiftDown();
4. [78, 60, 33, 39, 28, 66, 70] swap(heap[0], heap[2]);
5. [78, 60, 70, 39, 28, 66, 33] swap(heap[2], heap[6]);

b. É inserido um elemento com a prioridade 50 (no vetor original).

1. Vetor [95, 60, 78, 39, 28, 66, 70, 33];
2. [95, 60, 78, 39, 28, 66, 70, 33, 50] heap.append(50);
3. Entra na função iterativa shiftUp();
4. [95, 60, 78, 50, 28, 66, 70, 33, 39] swap(heap[8], heap[3]);

3) Escreva uma função que verifica se um vetor armazena um heap de máximo. A função deve receber um vetor como entrada e devolver true caso seja heap-max ou false caso contrário.

4) Implemente uma **fila de prioridade que use um Heap**. Se baseie nos algoritmos vistos no slide da aula.