

1- Escreva o algoritmo de inserção de uma chave x em uma **lista duplamente encadeada, ordenada, circular com nó cabeça**.

2- Considere uma árvore binária com raiz apontada por ptr_{raiz} , onde cada nó contém campos esq , dir , $chave$, e $soma_prefixo$;

2.1- Escreva um algoritmo eficiente que preencha, para cada nó da árvore o campo $soma_prefixo$ (que deve conter a soma de todas as chaves menores ou iguais à chave do nó que estão na árvore).

2.2- Daniel diz que conhece um algoritmo que insere chaves em uma árvore que já tem o campo $soma_prefixo$ preenchido, e que atualiza este campo para os nós da árvore em tempo $O(h)$, onde h é a altura da árvore. O algoritmo de Daniel pode estar correto? Justifique.

3 - Insira em uma árvore AVL inicialmente vazia, as seguintes chaves, nesta ordem (Mostre a árvore antes e após qualquer rotação!)

3.1 - 32, 10, 15, 25, 30.

3.2 - (na árvore de 3.1, insira) 19, 18, 17 e 20.

4 – Insira em uma árvore rubro-negra left-leaning inicialmente vazia, nesta ordem (Mostre a árvore antes e após qualquer rotação!)

4.1 - 32, 10, 15, 25, 30

4.2 - (na árvore de 4.1) 19, 18, 17, 20.

5- Considere uma heap binária de Mínimo correspondente ao vetor abaixo.

5.1 - Desenhe a representação em árvore desta heap binária.

5.2- Insira nesta heap, as chaves 20, 7, 11 e 18, e mostre a heap resultante.

5.3- Da heap original (de 5.1), remova a chave mínima.

5	8	15	19	10	23						
---	---	----	----	----	----	--	--	--	--	--	--