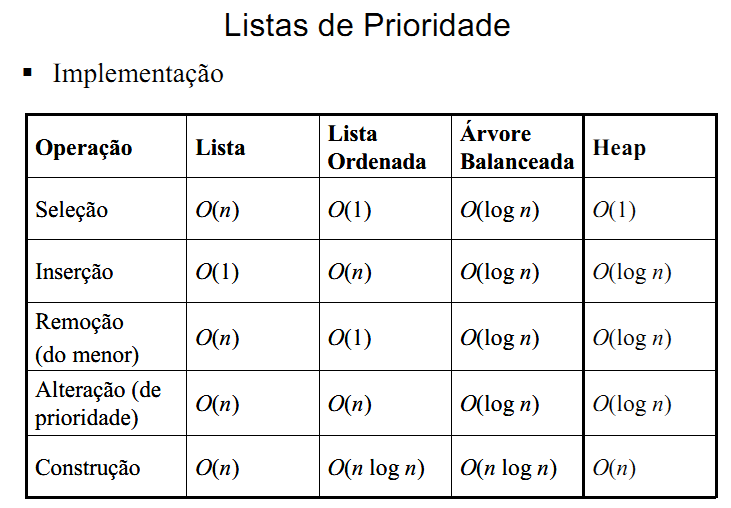
**Heaps**

**Consultar pdf:** EstrDadosAlgoritmos-ClaudioEsperanca.pdf , 0\_Inicio\_aula16

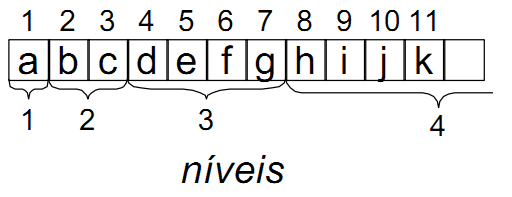
1) Heaps são estruturas usadas em Listas de Prioridade e HeapSort. A lista de prioridade pode ser implementada usando Lista, Lista Ordenada, Árvore balanceada e Heap. Faça um quadro comparativo de pior caso para as operações: Seleção, Inserção, Remoção, Alteração (de prioridade) e Construção, para cada uma das implementações citadas.



2) Diferentemente de árvores binárias de busca, heaps são implementados usando arrays.

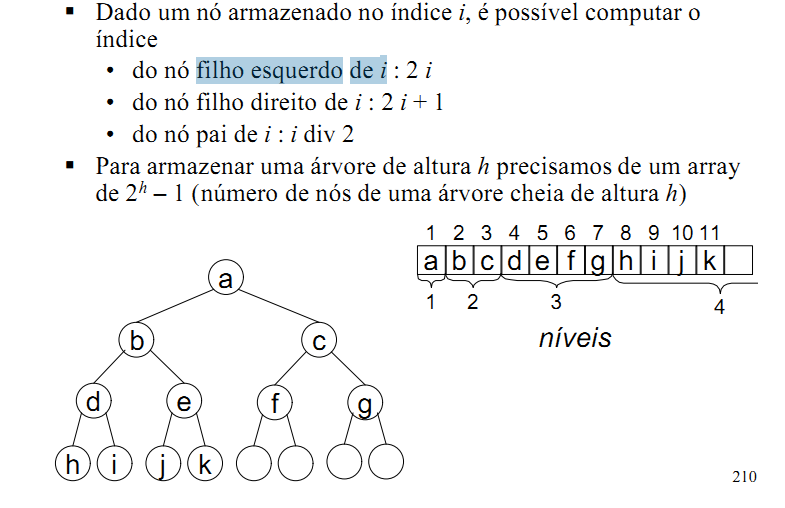
2.1) Para a figura abaixo, onde o vetor começa em 1, determine, mostrando como fez para chegar a solução:

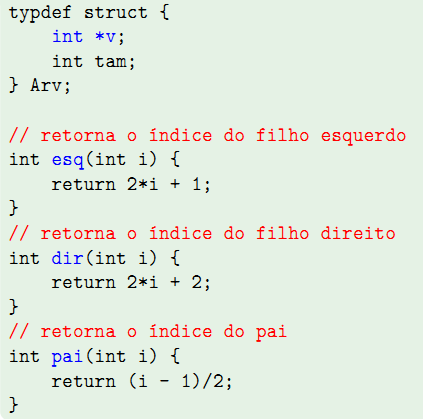
1. Quem é o pai do elemento com índice 5 (e)
2. Quem é o filho esquerdo do elemento com índice 5(e)
3. Quem é o filho direito do elemento com índice 5(e)



2.2) Agora supondo que este vetor comece em zero, determine, mostrando como fez para chegar a solução:

1. Quem é o pai do elemento d
2. Quem é o filho esquerdo do elemento d
3. Quem é o filho direito do elemento d





3) Qual a caraterística de um Heap de Máximo e de Mínimo.

5) Se um nó tem seu valor alterado, a manutenção das propriedades do Heap pode requerer que nó migre na árvore:

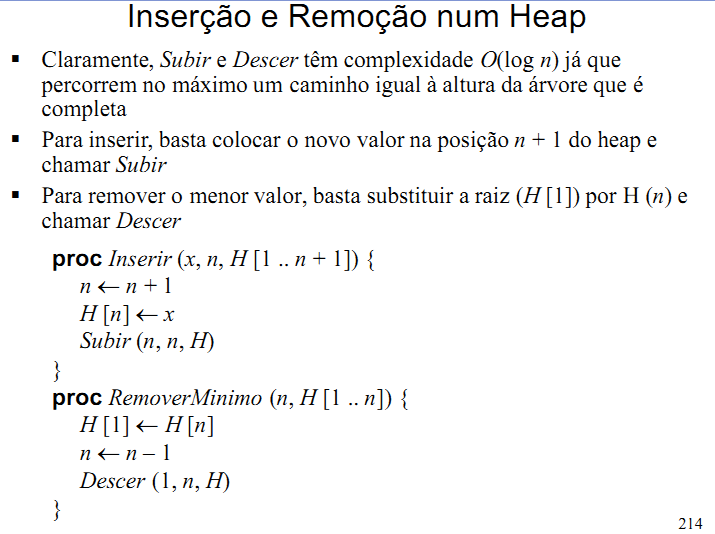
* para cima (se ele diminuir de valor)
* para baixo (se ele aumentar de valor)

Para cada uma dessas situações utiliza-se um algoritmo de migração:

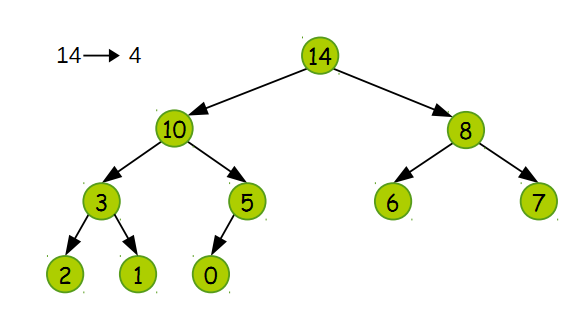
* subir (i, n, H) migra o nó i para cima no heap H
* descer (i,n,H) migra o nó i para baixo no heap H (sendo n o número total de nós da árvore/heap)

Para inserir, basta colocar o novo valor na posição n + 1 do heap e chamar Subir.

Para remover o menor valor, basta substituir a raiz (H [1]) por H (n) e chamar Descer.

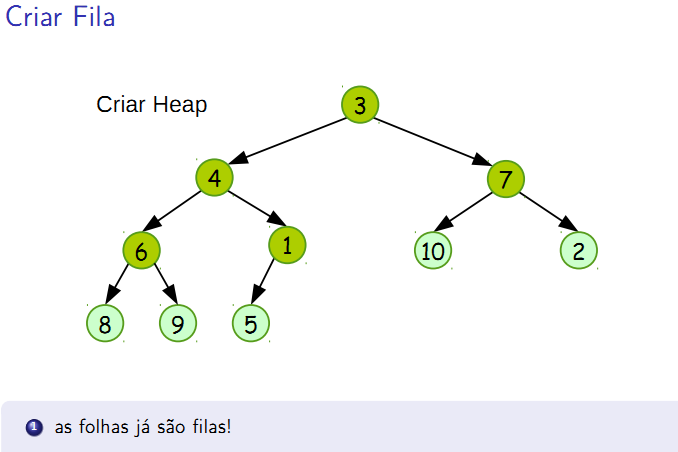


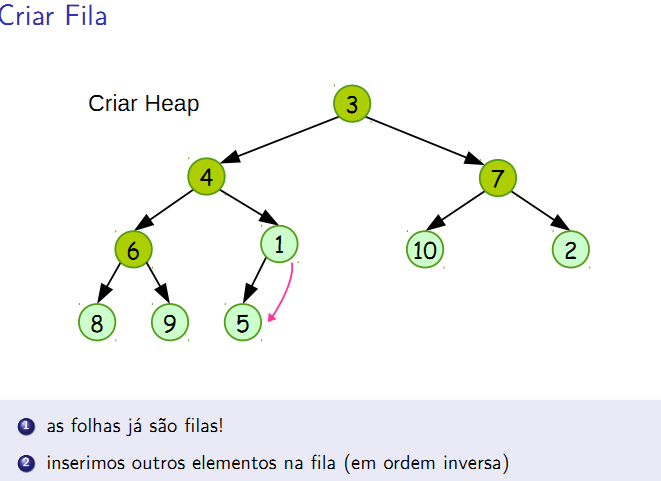
6) Supondo que a prioridade do nó raiz foi diminuída de 14 para 4, faça as operações para restabelecer a propriedade do heap:

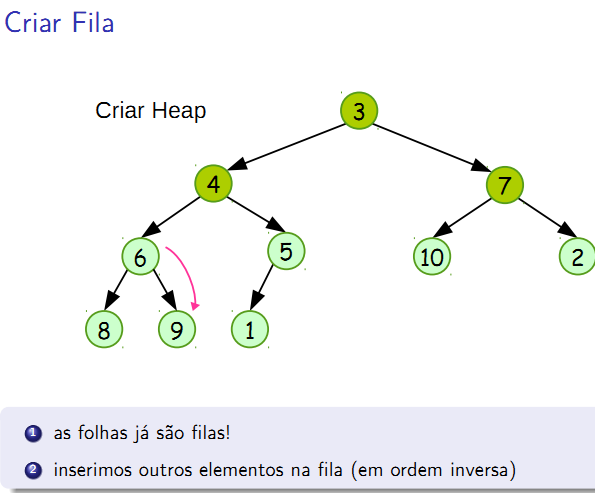


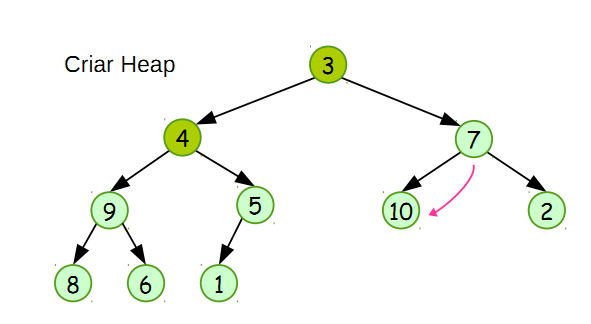
9) Dado o vetor abaixo, construir um heap de Maximo, executando passo a passo o algoritmo esperto de inserção. Supor que o vetor começa na posição 1 (3) .

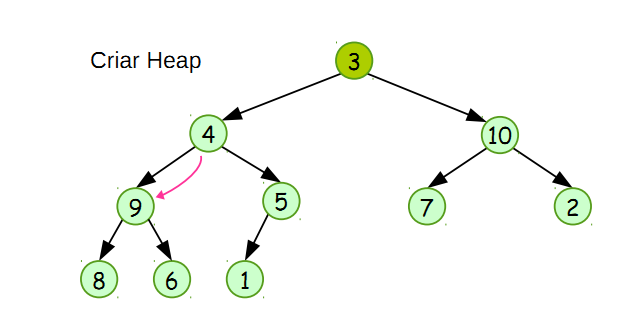
|3|4|7|6|1|10|2|8|9|5|

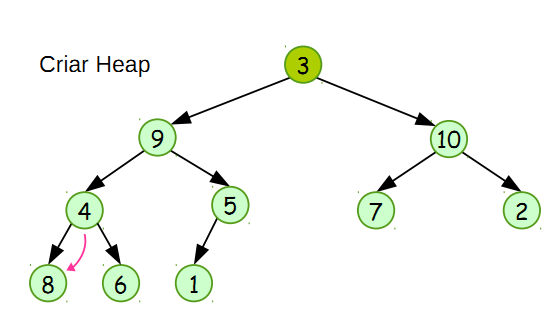


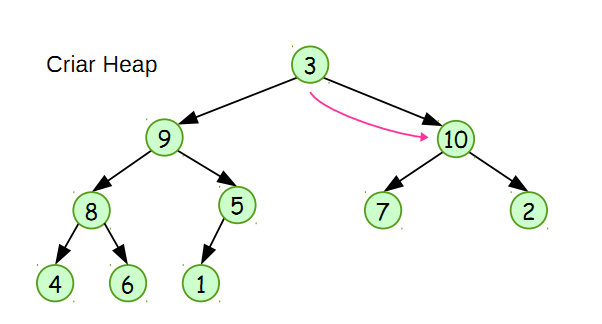


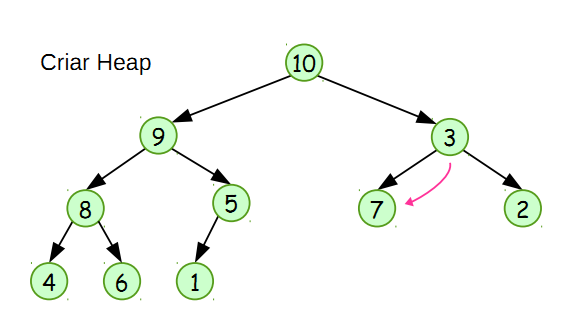


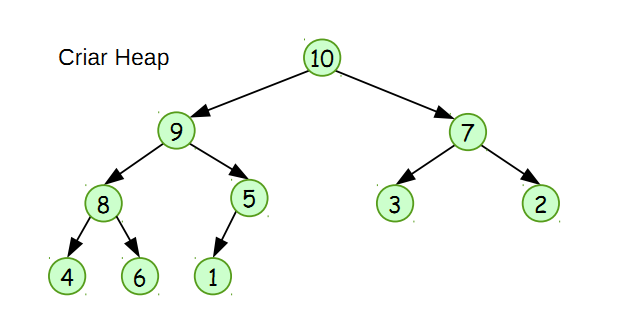




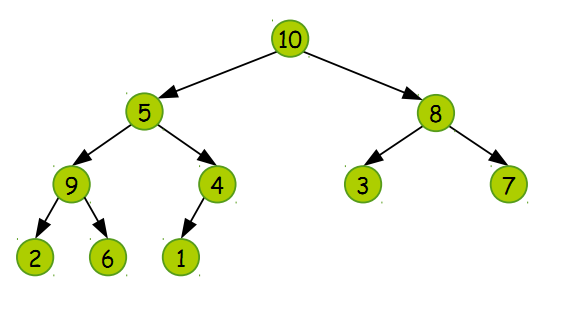


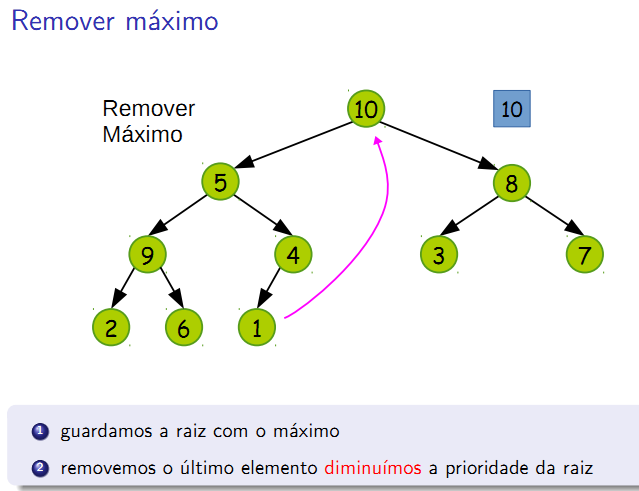






10) Remover a raiz (o elemento de maior prioridade) e rebalancear o heap. Descreva como fará o procedimento e o execute sobre o heap abaixo, passo a passo.





11) Descreva ou implemente um algoritmo para o HeapSort:

