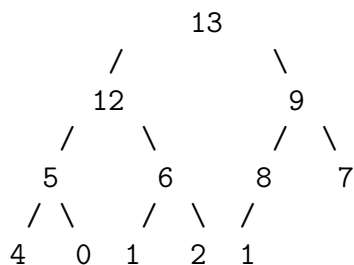


Universidade Tecnológica Federal do Paraná (UTFPR)  
 Departamento Acadêmico de Informática (DAINF)  
 Professora: Juliana de Santi  
**Estruturas II**

## Lista de exercícios

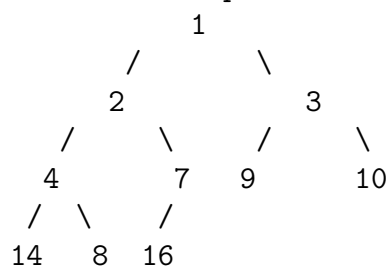
**Cormen)** Ilustre as operações de HEAP-EXTRACT-MAX e MAX-HEAP-INSERT( $V, size, 10$ ), de forma independente, sobre o heap  $V = \langle 15, 13, 9, 5, 12, 8, 7, 4, 0, 6, 2, 1 \rangle$  (mostre uma árvore para cada operação).

Mostre como implementar uma operação HEAP-DECREASE-KEY para um heap máximo. Para testar sua função use a entrada  $V = \langle 15, 13, 9, 5, 12, 8, 7, 4, 0, 6, 2, 1 \rangle$ , alterando a chave da posição zero para o valor 1, a qual deve gerar a seguinte saída:

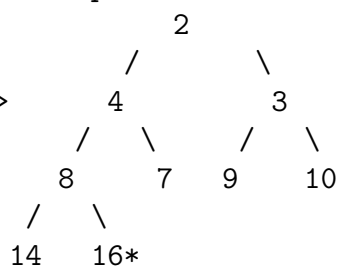


**Cormen)** Escreva um programa em C que implemente as operações de BUILD-MIN-HEAP, MIN-HEAPIFY, HEAP-MINIMUM, HEAP-EXTRACT-MIN, HEAP-DECREASE-KEY e MIN-HEAP-INSERT que implementam uma fila de prioridade mínima com um heap mínimo. Como exemplo suponha o vetor  $V = \langle 4, 1, 3, 2, 16, 9, 10, 14, 8, 7 \rangle$ . As operações BUILD-MIN-HEAP, HEAP-EXTRACT-MIN, HEAP-DECREASE-KEY( $V, 8, 1$ ) e MIN-HEAP-INSERT( $V, size, 0$ ), nesta sequência, produzem como resposta os seguintes heaps mínimos:

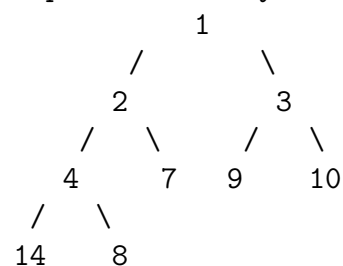
Build-Min-Heap:



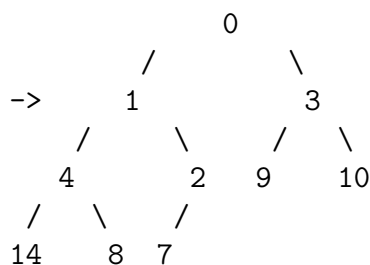
Heap-Extract-Min:



Heap-Decrease-Key(16->1)



Min-Heap-Insert (chave: 0)



Os exercícios a seguir (\*) são exercícios extra, ou seja, não fazem parte da lista para pontuação. Não haverá lista-solução e o estudante que quiser resolvê-los, deverá pensar a solução por conta própria.

**Exercício \*)** Um heap (fila de prioridade) é uma estrutura de dados muito importante, que tem duas utilidades principais: organizar acesso a um recurso com base na prioridade dos requerentes (processos, impressões, etc.) ou servir como base a um algoritmo de ordenação muito eficiente denominado heapsort. Para poder servir a esses propósitos, um heap possui uma série de propriedades especiais que têm que ser mantidas por todas as operações nelas realizadas. Levando em consideração estas propriedades, analise as afirmativas abaixo.

I)  $|50|40|49|39|45|46|$ : representa um heap máximo sintaticamente correto.

II) Dado o heap  $|21|14|10|9|5|$ , a inserção do elemento 12 se dá pelos passos  $|21|14|10|9|5|12| \rightarrow |21|14|12|9|5|10|$

III) Dado o heap  $|21|14|10|9|5|$ , a retirada do elemento do topo se dá através dos passos  $|5|14|10|9| \rightarrow |14|5|10|9| \rightarrow |14|10|9|5|$

**Exercício \*)** As filas de prioridades (heaps) são estruturas de dados importantes no projeto de algoritmos. Em especial, heaps podem ser utilizados na recuperação de informação em grandes bases de dados constituídos por textos. Basicamente, para se exibir o resultado de uma consulta, os documentos recuperados são ordenados de acordo com a relevância presumida para o usuário. Uma consulta pode recuperar milhões de documentos que certamente não serão todos examinados. Na verdade, o usuário examina os primeiros  $m$  documentos dos  $n$  recuperados, em que  $m$  é da ordem de algumas dezenas.

Considerando as características dos heaps e sua aplicação no problema descrito acima, avalie as seguintes afirmações.

I) Uma vez que o heap pode ser interpretado como uma árvore binária de pesquisa quase completa, o custo computacional para sua construção é  $O(n \log n)$

II) A implementação de heaps utilizando-se vetores é eficiente em tempo de execução e em espaço de armazenamento, pois o pai de um elemento armazenado na posição  $i$  se encontra armazenado na posição  $2i+1$

III) O custo computacional para se retirar de forma ordenada os  $m$  documentos mais relevantes armazenados em um heap de tamanho  $n$  é  $O(m \log n)$

IV) Determinar o documento com maior valor de relevância armazenado em um heap tem custo computacional  $O(1)$

**Exercício \*)** Qual o número mínimo (e o máximo) de prioridades (elementos) em uma fila de prioridades (heap máximo) de altura  $h$ ?

**Exercício \*)** Onde está situada a menor prioridade(elemento) em uma fila de prioridades (heap máximo)?

**Exercício \*)** Qual é a relação entre um min-heap e um array ordenado?

**Exercício \*)** Um arranjo ordenado em ordem decrescente é um heap máximo? Um heap máximo é um arranjo ordenado em ordem decrescente?