Universidade Federal da Bahia
Departamento de Ciência da Computação

Heap

Ricardo Araújo Rios

Introdução

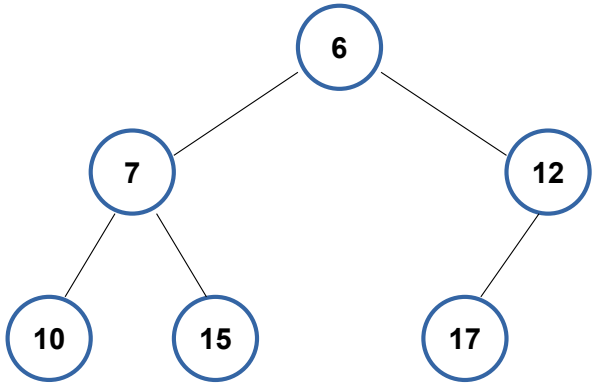
Introdução

- Heap é uma estrutura de dados organizada como uma árvore binária;
- Para uma árvore binária ser considerada uma heap, algumas regras precisam ser satisfeitas:
  - Ordem dos nós;
  - Completude;

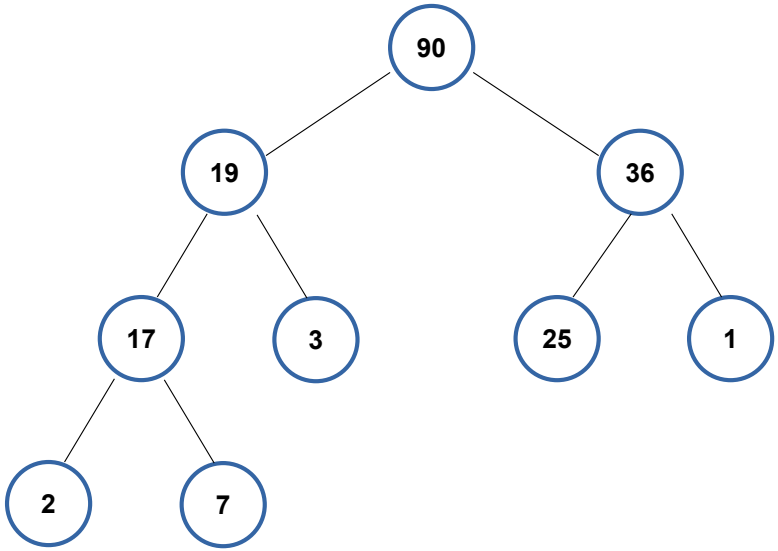
Introdução

- Regra 01 – Ordem dos nós:
  - Para cada nó **v** da árvore, exceto a raiz, temos:
    - chave(pai(v)) ≤ chave(v) – min-heap
    - chave(pai(v)) ≥ chave(v) – max-heap

Min-heap



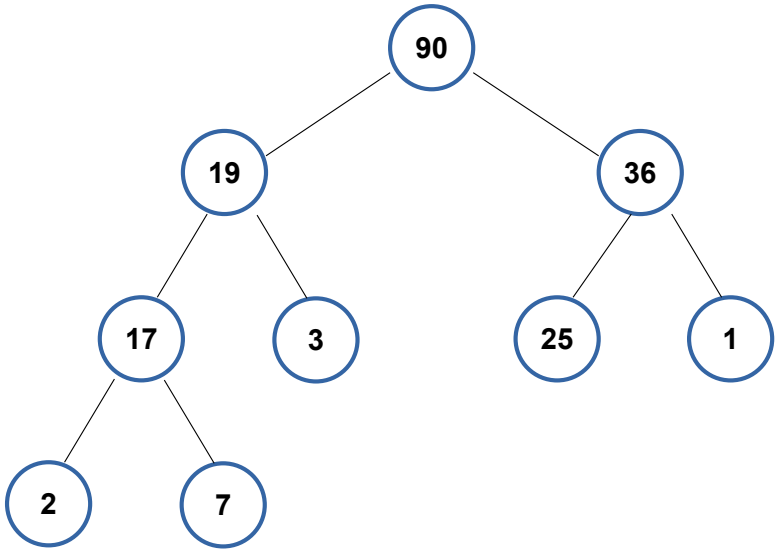
Max-heap



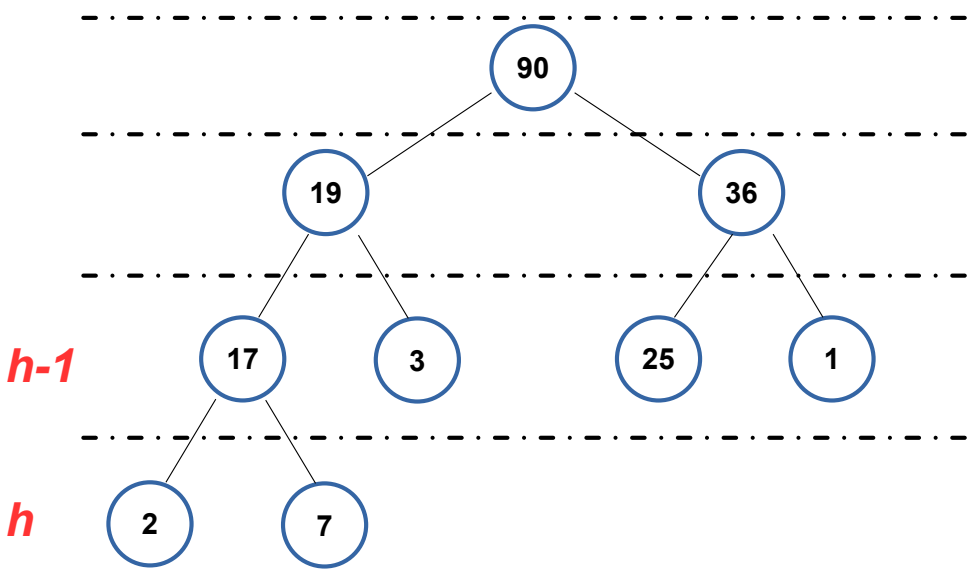
Introdução

- Regra 02 – Completude (balanceamento)
  - A estrutura é dita ser completa, ou está balanceada, se:
    - Todo nó folha está ou no nível h ou h-1;
    - O nível h-1 deve estar completamente preenchido;
    - Se o nível h não estiver preenchido completamente, as folhas estão mais à esquerda.

Exemplo



Exemplo



Operações Básicas

- Inserção
  - Fix-up
- Remoção
  - Fix-down

Árvore Binária de Pesquisa

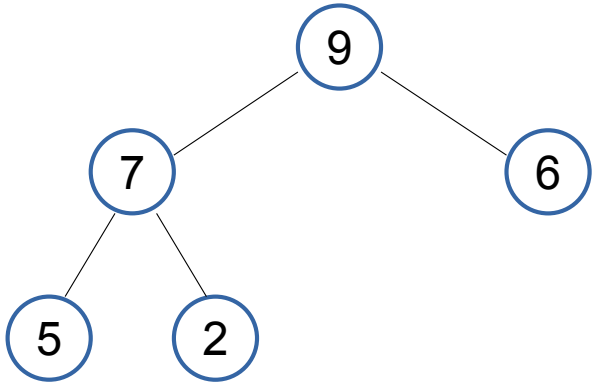
Inserção

Inserção

- Inserir o novo item no final da heap;
- Restaurar a ordem da heap (fix-up).

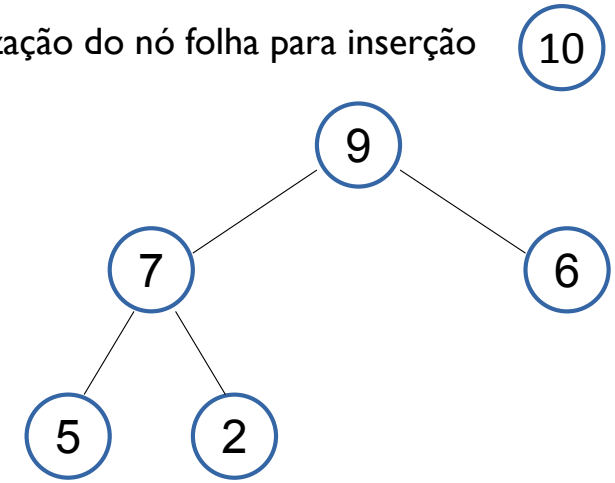
Inserção

- Localização do nó folha para inserção



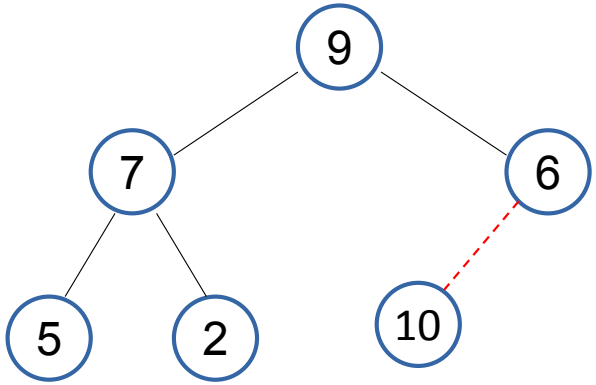
Inserção

- Localização do nó folha para inserção



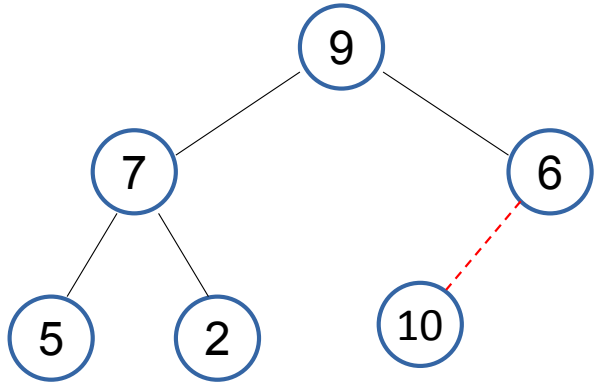
Inserção

- Localização do nó folha para inserção



Inserção

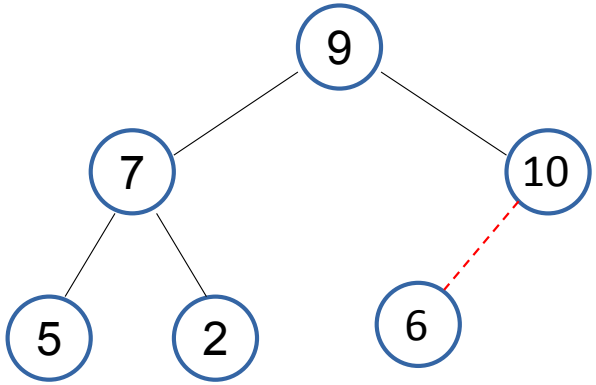
- Localização do nó folha para inserção



Realiza troca se o filho for maior que o pai

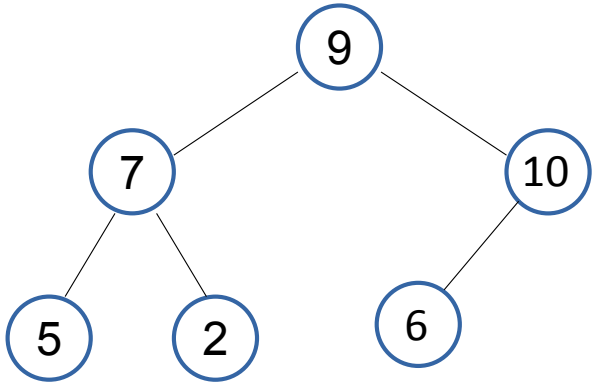
Inserção

- Localização do nó folha para inserção



Inserção

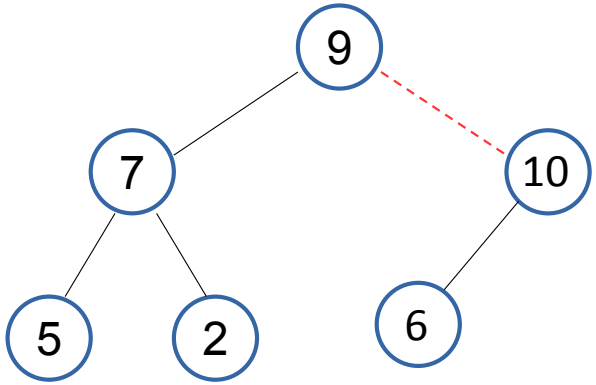
- Localização do nó folha para inserção



Realiza troca se o filho for maior que o pai

Inserção

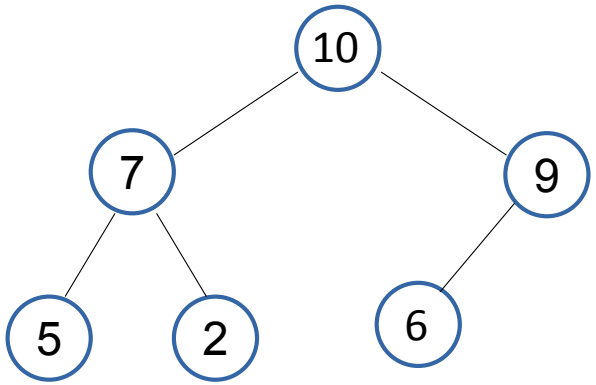
- Localização do nó folha para inserção



Realiza troca se o filho for maior que o pai

Inserção

- Localização do nó folha para inserção

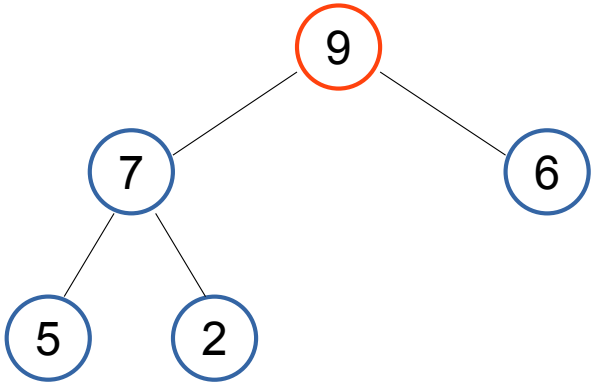


Critério de parada: nenhum pai maior que o elemento inserido ou quando o elemento inserido chegar à raiz.

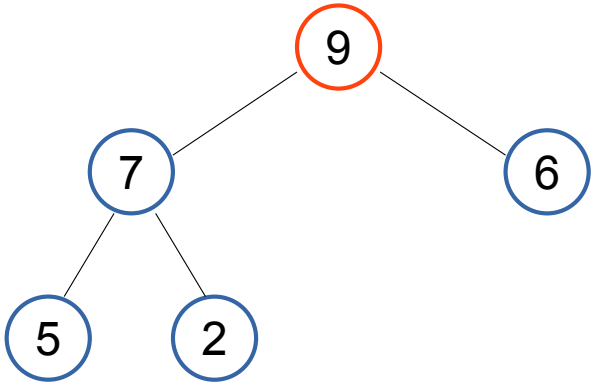
Remoção

- Substitui o nó a ser removido pelo último nó da heap;
- Restaurar a ordem da heap (fix-down).

Remoção

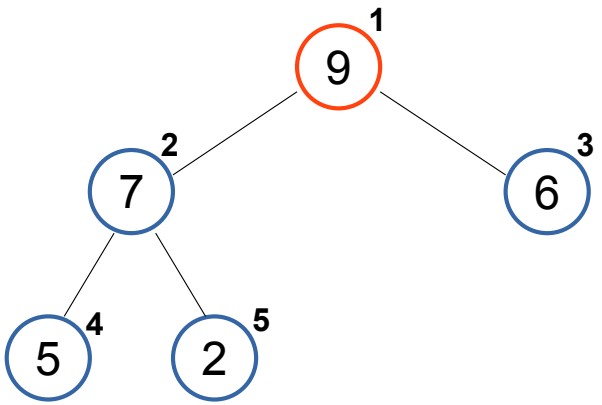


Remoção



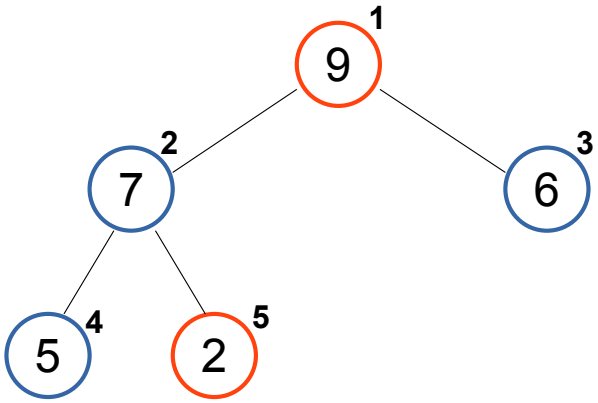
Seleciona o último elemento

Remoção



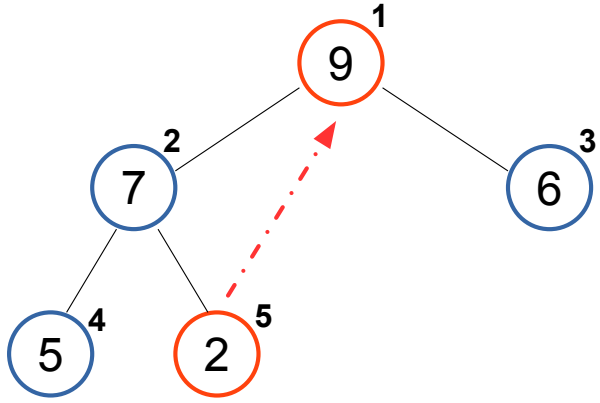
Seleciona o último elemento

Remoção

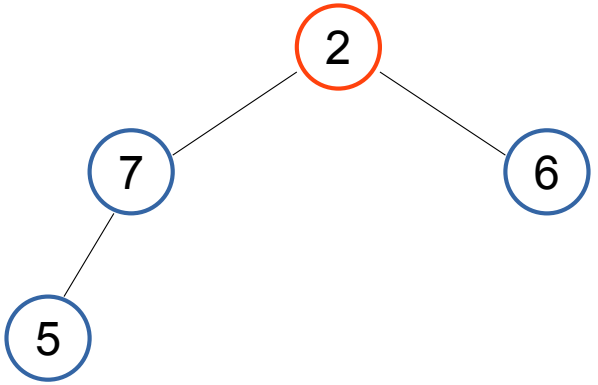


Seleciona o último elemento

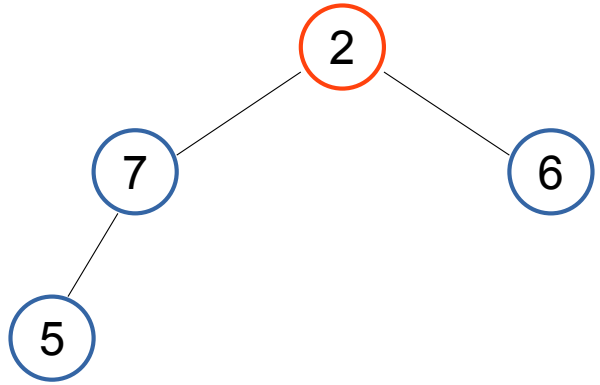
Remoção



Remoção

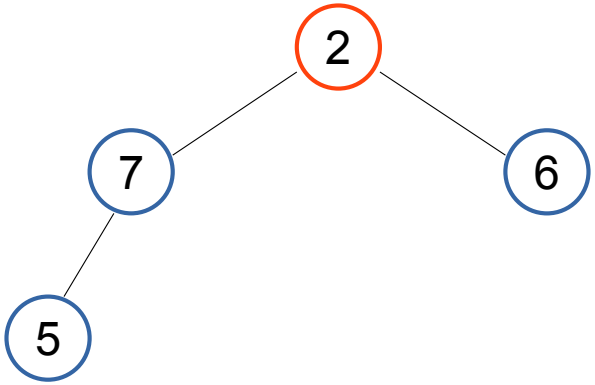


Remoção



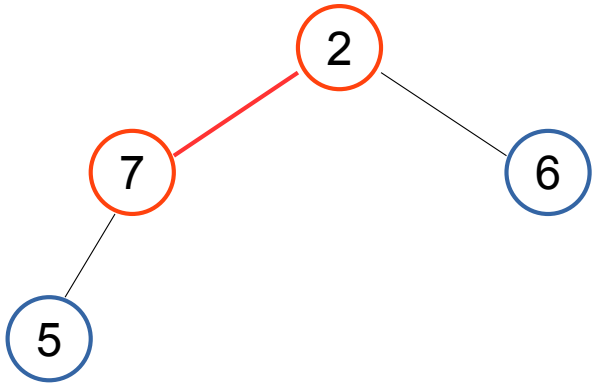
Fix-down!

Remoção



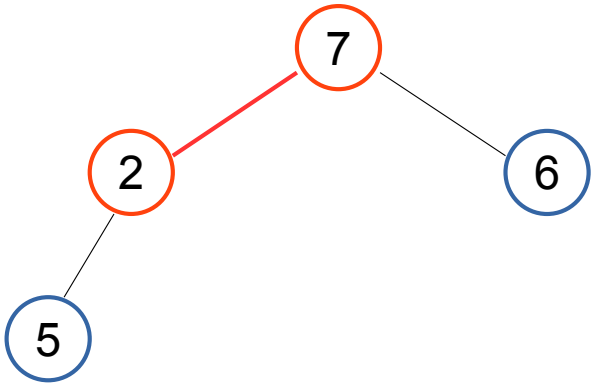
Se pai for menor que os filhos, troca o pai pelo filho com maior valor

Remoção



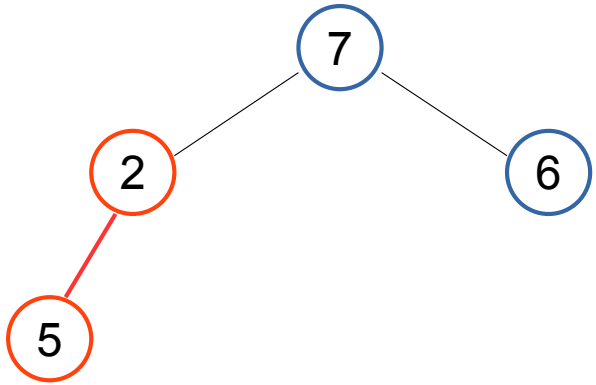
Se pai for menor que os filhos, troca o pai pelo filho com maior valor

Remoção



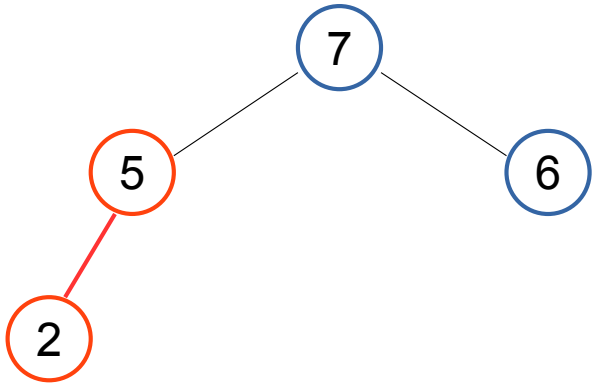
Se pai for menor que os filhos, troca o pai pelo filho com maior valor

Remoção



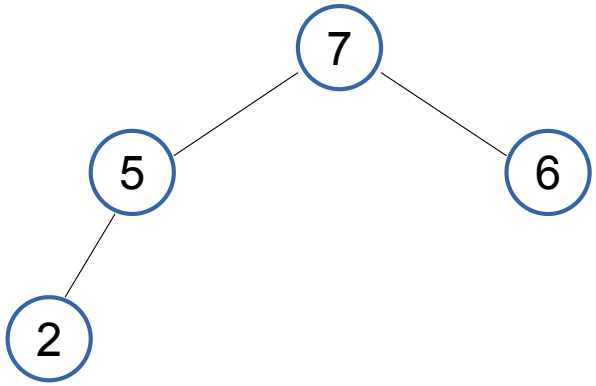
Se pai for menor que os filhos, troca o pai pelo filho com maior valor

Remoção



Se pai for menor que os filhos, troca o pai pelo filho com maior valor

Remoção



Se pai for menor que os filhos, troca o pai pelo filho com maior valor

Utilização

- Fila de prioridade
- Exemplo: supondo que quanto maior o valor, maior a prioridade, a remoção de um elemento é realizada sempre removendo o elemento raiz e restaurando a ordem da heap (fix-down)

Utilização

- Por que usar Heap?

TAD	Operação	Tempo
Lista não-ordenada	Inserir	O(1)
	Remover	O(n)
Lista ordenada	Inserir	O(n)
	Remover	O(1)
Heap	Inserir	O(log n)
	Remover	O(log n)

Referências

[1] Cormen, T. H., Leiserson, C. E., Rivest, R. L., Stein, C., Algoritmos – Teoria e Prática, 2ª Edição, Elsevier, 2002;

[2] Kleinberg, J., Tardos, E., Algorithm Design, Pearson, 2006;

[3] Goodrich, M. T., Tamassia, R., Estrutura de Dados e Algoritmos em Java, 4ª Edição, Bookman, 2007;

[4] Ziviani, N., Projeto de algoritmos com implementações em Java e C++, Thomson, 2007