

Universidade Federal Fluminense
Instituto de Computação
Curso de Sistemas de Informação
Estruturas de Dados para SI

Segunda Lista de Exercícios

Árvores

- 1) Escreva um programa em C que converta uma árvore genérica, descrita por um array de filhos associado a cada nó, em uma floresta de árvores binárias filho-esquerdo/irmão-direito.
- 2) Escreva um programa em C que faça o inverso do exercício 1.
- 3) Mostre que toda árvore pode ser representada por uma sequência binária de 0s e 1s.

Árvores binárias

- 1) Sejam T_0 e T_1 duas árvores binárias. Escreva uma função recursiva em C que verifique se T_0 e T_1 são idênticas.
- 2) Escreva uma função em C que percorra uma árvore binária em largura.
- 3) Escreva uma função em C que percorra uma árvore binária em ordem.
- 4) Escreva uma função em C que verifique se uma árvore binária é completa.
- 5) Sejam T_1 e T_2 duas árvores binárias de busca. Escreva uma função em C que concatene as duas árvores em uma única.
- 6) Escreva um algoritmo que retorne o maior elemento de uma árvore binária de busca.
- 7) Escreva um algoritmo que retorne o sucessor de uma dada chave em uma árvore binária de busca.

Árvores balanceadas

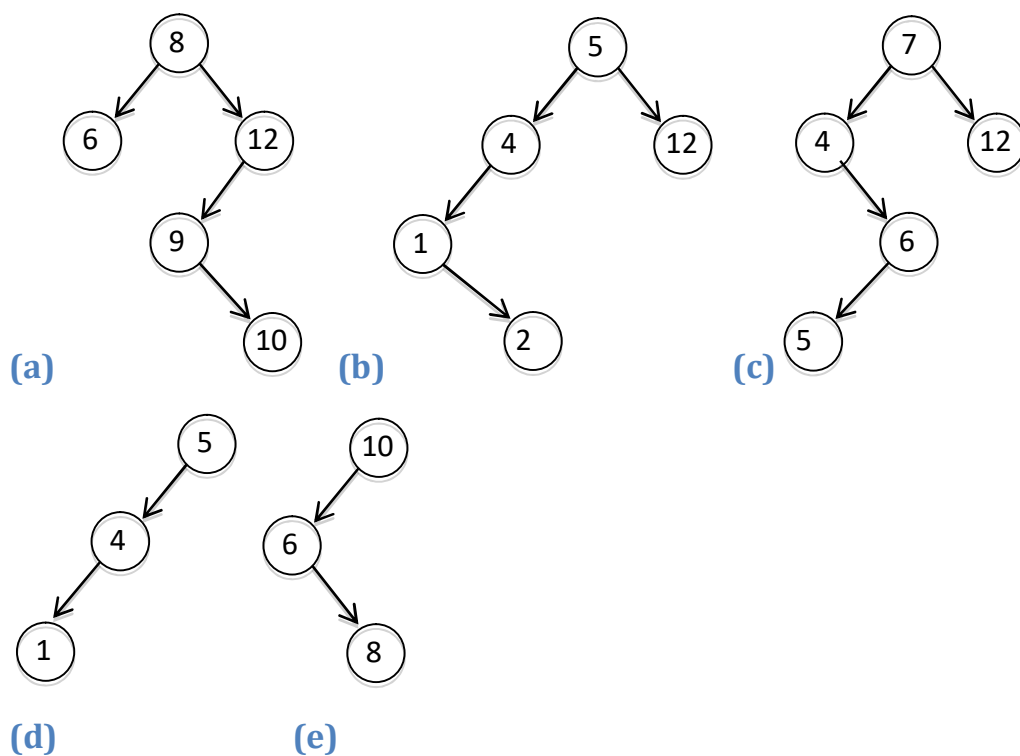
- 1) Construa a árvore AVL correspondente a inserção das chaves 9,15,1,2,4,23,17,8 na ordem dada.
- 2) Descreva em pseudo-código o algoritmo de inserção de um nó em uma árvore AVL.

3) Descreva os algoritmos de rotação direita, rotação dupla direita, rotação-esquerda e rotação dupla esquerda.

4) Mostre que a rotação dupla esquerda pode ser obtida por uma rotação direita seguida de uma rotação esquerda.

5) Mostre que uma rotação dupla direita pode ser obtida por uma rotação dupla esquerda seguida de uma direita.

6) Analise os casos abaixo considerando uma árvore AVL e identifique o tipo de transformação de balanceamento que deve ser efetuado. Mostre o resultado após o balanceamento.



Árvores B

1) Desenhe graficamente a árvore b de ordem 1 após a:

- Inserção das chaves 4,8,32,45,12,9,7,23,11,10,1
- Remoção das chaves 12,1,32,9,23

2) Descreva, em pseudo-código, o algoritmo para cisão de uma página em uma árvore B.

3) Descreva, em pseudo-código, o algoritmo para concatenação de duas páginas de uma árvore B.

4) Descreva, em pseudo-código, o algoritmo para redistribuição de chaves entre duas páginas de uma árvore B.

Listas de Prioridade

- 1) Com base em uma estrutura de heap implemente o algoritmo HeapSort.
- 2) Mostre como obter o segundo maior valor de um heap em tempo constante.
- 3) Faça um algoritmo que retorne o k-ésimo maior valor de um heap.
- 4) Através de um array implemente uma árvore binária completa d-ária.

Hashing

- 1) Demonstre o que ocorre após inserir as chaves 5; 28; 19; 15; 20; 33; 12; 17; 10 em uma tabela de dispersão com colisões tratadas por endereçamento externo. Suponha que a tabela tenha 9 containers e a função de dispersão seja dada por $h(x) = x \bmod 9$.
- 2) Escreva um programa em C que implemente uma tabela de dispersão com endereçamento exterior.
- 3) Escreva um programa em C que implemente uma tabela de dispersão com endereçamento aberto duplo.