

Estruturas de Dados: Lista 04

Marcelo Hashimoto

Última Atualização: 5 de março de 2015

Nesta lista, você **sempre** pode supor que os nós pertencem a *árvores binárias de inteiros*, mas **nem sempre** pode supor que essas árvores são *de busca binária*.

Nos quatro exercícios abaixo, você **não pode** supor que as árvores são de busca binária.

1. Escreva uma função **recursiva** que recebe o endereço de um nó e devolve o *menor valor* da subárvore enraizada nesse nó.
2. Escreva uma função **recursiva** que recebe o endereço de um nó e devolve o *maior valor* da subárvore enraizada nesse nó.
3. Escreva uma função **recursiva** que recebe o endereço de um nó, devolve 1 se a subárvore enraizada nesse nó for de busca binária e devolve 0 caso contrário. Utilize as funções dos Exercícios 1 e 2.
4. **[DESAFIO]** Utilizar as funções dos Exercícios 1 e 2 para determinar se uma árvore satisfaz a propriedade de busca binária é ineficiente. Implemente um algoritmo melhor.

Nos seis exercícios abaixo, você **pode** supor que as árvores são de busca binária.

5. Escreva uma função **recursiva** que recebe o endereço de um nó e devolve o endereço do *mínimo* da subárvore enraizada nesse nó. Você pode supor que o endereço recebido não é NULL.
6. Escreva uma função **iterativa** que recebe o endereço de um nó e devolve o endereço do *mínimo* da subárvore enraizada nesse nó. Você pode supor que o endereço recebido não é NULL.
7. Escreva uma função que recebe o endereço de um nó e devolve o endereço do *sucessor* desse nó. Caso esse sucessor não exista, devolve NULL. Você pode supor que o endereço recebido não é NULL.
8. Escreva uma função que recebe o endereço de uma árvore e um inteiro e insere nessa árvore um nó cujo valor é esse inteiro.
9. Escreva uma função que recebe o endereço de uma árvore e um inteiro e retira dessa árvore um nó cujo valor é esse inteiro. Caso o valor de nenhum nó seja esse inteiro, a função não faz nada. Caso o valor de múltiplos nós seja esse inteiro, a escolha pode ser arbitrária.
10. Escreva um programa que pede para o usuário digitar um inteiro positivo n , inicializa uma árvore vazia e realiza uma sequência de n inserções aleatórias, seguida por uma sequência de n retiradas aleatórias. Após cada operação, utilize as funções dos Exercícios 3 ou 4 para verificar se a árvore está correta.