Faculdade Paraíso do Ceará
SI014 – Algoritmos e Estruturas de Dados II
Prof. Vicente H. F. Batista

Lista de Exercícios 1B

entregar até 11 de outubro de 2011

Parte Teórica

Questão 1 (CLRS, Exercício 12.3-3)**.**

Questão 2. Dada uma árvore binária T, elabore um algoritmo para testar se T é uma árvore AVL. Qual a complexidade de seu algoritmo?

Questão 3 (CLRS, Exercício 13.1-3)**.**

Questão 4 (CLRS, Exercício 13.1-4)**.** Suponha que todo nó vermelho de uma árvore rubro-negra seja "absorvido" por seu pai preto, de modo tal que os filhos do nó vermelho se tornem filhos de seu pai preto. Quais são os valores possíveis para o grau de um nó preto após ter absorvido seus filhos vermelhos? O que você poderia dizer sobre as profundidades das folhas da árvore resultante?

Questão 5. A *colocação* de um nó x em uma árvore binária de busca T é igual ao número de nós com chave menor do que $x.chave$ mais 1. Elabore um algoritmo recursivo denominado COLOCAÇÃO(k, x) que retorna a colocação do nó em T com chave igual a k. Dica: utilize a função TAMANHO(x), a qual retorna o número de nós da subárvore de T com raiz em x.

Parte Prática

Questão 6. Suponha que você foi encarregado de implementar um novo filtro para o site Balcão do NexTI da FAP (`http://nexti.fapce.com.br/balcao/`). Este filtro deverá selecionar todos os produtos com preços variando entre a e b, inclusive. Supondo que os produtos do Balcão estão armazenados em uma árvore AVL, implemente esta ideia em uma função cujo cabeçalho é o seguinte:

```
void busca_por_intervalo(Node *x, Key a, Key b, Node **s);
```

onde `x` é um nó da árvore, `a` e `b` são os limites do intervalo de busca e `s` é um arranjo de objetos do tipo `Node *`, suficientemente grande. Utilize o código trabalhado em sala de aula como ponto de partida.