

Exercícios - Lista 6

INSTRUÇÕES: A solução completa da lista de exercícios deve ser enviada, **em arquivo Scheme**, para o e-mail `lmduarte@inf.ufrgs.br` **até 09/06/2009**. O assunto do e-mail deve seguir o formato **"Lista 6 - [Nome]"**, onde [Nome] corresponde ao nome do aluno. Cada resposta deve ser identificada quanto à questão a que se refere. **Respostas enviadas sem seguir estas instruções poderão ser ignoradas.**

1. Considere a seguinte definição de dados:

```
(define-struct nó (val esq dir))
```

Uma AB (árvore binária) é ou
- false, ou
- (make-nó val esq dir),
onde val é número, esq e dir são AB

Defina a função `valores-AB` que, dada uma árvore binária de entrada, retorna uma lista com os valores de todos os nós da árvore (em qualquer ordem).

2. Escreva uma função `acha-maior-AB` que encontre o maior valor armazenado em uma árvore binária.
3. Apresente um programa `converte-árvores`, o qual, dada uma AB, converte esta estrutura em uma ABP.
4. Uma página web é:

- (a) `empty`, ou
- (b) `(cons s wp)`, onde `s` é um símbolo e `wp` é uma página web, ou
- (c) `(cons ewp wp)`, onde `ewp` e `wp` são páginas web

A partir dessa definição de dados, desenvolva a função `troca`. A função recebe dois símbolos, `novo` e `velho`, e uma página web `wp`. A função produz uma página com a mesma estrutura de `wp`, mas com todas as ocorrências de `velho` trocadas por `novo`.

5. Dadas as definições vistas em aula para uma árvore genealógica, a estrutura `parent` e uma `lista-de-filhos`, desenvolva a função `conta-descendentes`, a qual conta o número de descendentes de um nó, incluindo o próprio nó.
6. **DESAFIO:** Crie um programa que remova um dado nó de uma ABP, sem alterar o critério de menores à esquerda e maiores à direita.