

INF 05008 - Fundamentos de Algoritmos 2020/2  
 Lista de exercícios 8  
 Estruturas com autorreferência (Cap. 14)

Siga as instruções sobre elaboração de exercícios de INF05008.

Além disso, nesta lista, para todas as **funções recursivas** você deve incluir modelo da solução. Esse modelo, em forma de comentários, deve explicitar como o algoritmo vai funcionar para o(s) caso(s) base e passo(s) da definição da função recursiva. Este modelo deve ter o seguinte formato, e pode ser colocado antes ou permeado ao código:

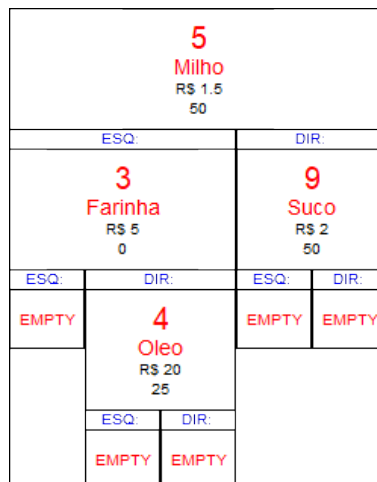
```
;; SE < é o caso base da def. de lista > ENTÃO < resolver o problema >
;; SE < não é o caso base da def. de lista > <=== (pode haver mais de um
                                                desses casos)

;; ENTÃO < combinar soluções >
;;      < fazer algo com > < o primeiro elemento da lista >
;;      < solucionar o problema para. > < o resto da lista >
```

Utilize o template fornecido no Moodle. Nele, existe uma função para a visualização para as árvores utilizadas nesta lista. Por exemplo, a árvore representada pelo seguinte código:

```
(make-nó-inv MILHO
  (make-nó-inv FARINHA
    empty
    (make-nó-inv OLEO empty empty))
  (make-nó-inv SUCO empty empty))
```

Será visualizada da seguinte forma:



## Exercícios

Na lista de exercícios 6, foi codificado um sistema para um mercado. Neste sistema, o inventário do mercado era representado por uma lista. A cada vez que é pesquisado o preço de um item, o sistema tem que pesquisar no inventário seu preço. Imagine um supermercado que tenha mais de 100 mil itens em seu inventário. No pior caso o sistema tem que verificar 100 mil produtos até encontrar o correto. Agora temos a oportunidade de melhorar esse sistema.

**Exercício 1.** Crie um novo tipo `Inventario-ABP` que representa um inventario, porém na forma de uma ABP. Para isso, crie uma estrutura chamada `nó-inv`. O conteúdo de cada nó deve ser um `Produto`, que já foi definido na lista 6, além de suas sub-árvores da direita e da esquerda. Seus atributos então devem ser: `produto`, `esq` e `dir`, nesta ordem. Use o código de barras do produto como identificador do nó. Use `empty` para representar um nó vazio. Defina uma constante desse novo tipo com, no mínimo, 5 produtos.

**Dica 1:** Para facilitar a obtenção do identificador de um nó, basta fazer uma função auxiliar que, dado um nó, retorna seu identificador.

**Exercício 2.** O dono do mercado não quer ter que inserir manualmente todos os itens do inventário antigo no novo inventário. Crie uma função `inventario-list-to-ABP` que, dado um `Inventario`, retorna um `Inventario-ABP` com todos os itens do inventário antigo.

**Exercício 3.** Crie uma função chamada `verifica-preco` que, dado um inventario em forma de ABP e um código de barras, retorna o preço do produto cadastrado no inventário com o código dado, com o desconto aplicado. Caso não haja um produto cadastrado com este código, retorna a string `"Produto não encontrado"`.

Agora copie a função `checkout`, codificada na lista 6, para o código atual. Substitua a chamada da antiga `verifica_preco`, pela chamada da função `verifica-preco`, criada agora. Faça isso sem alterar o resto do código (claro que agora a entrada deverá ser do tipo `Inventario-ABP`) e verifique se a função `checkout` continua funcionando.

**Exercício 4.** Faça uma função chamada `checa-descontos` que, dado um `Inventario-ABP` e um número `n` entre 0 e 100, retorna a quantidade de produtos que estão com um desconto maior que `n%`.

**Exercício 5.** Faça uma função chamada `lista-descontos` que, dado um `Inventario-ABP` e um número `n` entre 0 e 100, retorna uma lista dos produtos que estão com um desconto maior que `n%`.

**Exercício 6.** Faça uma função chamada `profundidade` que, dado um `Inventario-ABP`, retorna a profundidade da ABP dada. Lembre-se que a profundidade de uma árvore é dada pelo seu maior nível, que, por sua vez, é a distância de um nodo até a raiz. Se a árvore de entrada for vazia, deve retornar a string `"Não é possível calcular a profundidade de uma árvore vazia"`.