

# Teste de Programação Funcional

29 de Maio de 2013

DCC/FCUP

Versão A

Nome: \_\_\_\_\_

Nº mecanográfico: \_\_\_\_\_

- Este teste contém 4 questões e 2 páginas.
- Responda às questões 1 e 2 no espaço marcado no enunciado.
- Pode usar funções auxiliares e/ou do prelúdio-padrão de Haskell

## 1. (30%)

- (a) Escreva uma definição da função `nodups` que remove valores iguais consecutivos de uma lista. Exemplos:

```
nodups [1,1,1,2,1,3,3,4,4] = [1,2,1,3,4]
nodups "baannnana" = "banana"
```

- (b) Indique o tipo mais geral da função `nodups`.

2. (30%) Complete a seguinte definição apresentada nas aulas da inserção de um valor numa árvore binária de pesquisa.

```

insserir :: _____
insserir x Vazia = _____
insserir x (No y esq dir)
    | x==y = _____
    | x<y = _____
    | otherwise = _____

```

**Responda às questões seguintes numa folha de exame.**

3. (20%) Considere o seguinte tipo recursivo para representar texto formatado em “caixas” bi-dimensionais:

```

data Box = Text String    -- texto simples
        | Horiz Box Box  -- justaposição horizontal
        | Vert Box Box   -- justaposição vertical

```

Exemplo:

```
Horiz (Vert (Text "AA") (Text "BBBBB")) (Text "CCC")
```

representa a caixa 

AA	□□□	CCC
BBBBB		

 com largura de 8 carateres e altura de 2 linhas.

Pretende-se que escreva definições recursivas das funções

```
altura, largura :: Box -> Int
```

que calculam as dimensões exteriores duma caixa de texto.

4. (20%) Usando indução sobre listas, mostre que

$$\text{length } (\text{map } f \text{ } xs) = \text{length } xs$$

para quaisquer funções  $f$  e listas finitas  $xs$ . Tenha o cuidado de apresentar as definições das funções *length* e *map*.