

Nome: _____

Nº mecanográfico: _____

- Este exame contém 7 questões em 4 páginas e tem uma duração de 2h(+30m).
- Responda às questões no espaço marcado no enunciado.
- Pode usar funções auxiliares e/ou do prelúdio-padrão de Haskell.
- Nas questões 2 a 7, indique sempre o tipo da função definida.

1. (30%) Responda a cada uma das seguintes questões, indicando **apenas** o resultado de cada expressão.

(a) `length ([1,2]:[3]:[4,5]:[])` = _____

(b) `take 2 ([1,2]:[3]:[4,5]:[])` = _____

(c) `zipWith (+) [1,3..10] [10,9..1]` = _____

(d) `foldr (*) 2 [1,2,3,4,5]` = _____

(e) `takeWhile (>=3) [1,2,3,4,5]` = _____

(f) `(all (>3) . filter (>=3)) [1,2,3,4,5]` = _____

(g) `[(x^2,y) | (x,y) <- zip [1..6] [1,4..]]` = _____

(h) Defina a lista `[(1,10),(3,8),(5,6),(7,4),(9,2),(11,0),(13,-2),...]`:

(i) Considere a seguinte definição em Haskell:

```
h _ base [] = base
```

```
h op base (x:xs) = h op (op base x) xs
```

A avaliação da expressão `h (\x y -> x:y) [] [1,2,3,4,5]` tem como resultado:

(j) Indique um tipo admissível para a `(2*) . (+2)`:

(k) Indique um tipo admissível para `[map,filter]`: _____

(l) Considere as seguintes definições em Haskell:

```
data Arv a = ???
```

```
f :: Arv a -> [a]
```

```
f (F x) = [x]
```

```
f (N e d) = f e ++ f d
```

Complete a definição do tipo `Arv a`, para que a função `f` esteja bem definida:

(m) Indique o tipo mais geral de `g p xs = [x | x<-xs, p x]`:

2. (15%) Numa determinada disciplina, consideramos para cada aluno uma lista de notas (de 0 a 20) correspondendo a diferentes componentes da avaliação, assim como uma lista com os pesos (de 0 a 1) de cada componente na nota final. Por exemplo, as seguintes listas respectivas de notas e pesos $[11.5, 12, 15]$ $[0.4, 0.3, 0.3]$, indicam que a nota final será $11.5 \cdot 0.4 + 12 \cdot 0.3 + 15 \cdot 0.3$.

- (a) Implemente a função **notaF** que dadas uma lista de notas e a respectiva lista de pesos, calcula a nota final de um aluno.
- (b) Supondo que cada componente tem uma nota mínima de 8 valores, implemente a função **rfc** que dada uma lista (de listas) com as notas de todos os alunos, determina quantos alunos é que reprovaram por falta de componente.

*Nota: pode utilizar funções do prelúdio-padrão e/ou listas em compreensão mas não deve usar directamente **recursão**.*

3. (10%) Considere os seguintes tipos para representar grafos dirigidos:

```
type Vert = Int
type Graph = [(Vert, Vert)]
```

A relação expressa pelo grafo é transitiva se e só se, para todos os vértices v_1 , v_2 e v_3 , se (v_1, v_2) e (v_2, v_3) são arcos do grafo, então (v_1, v_3) tão é. Escreva uma função **transitiva** que verifica se um dado grafo representa uma relação transitiva.

4. (5%) A função **iterate** do prelúdio padrão, itera uma função à escolha do utilizador, a partir de um elemento. Por exemplo: **iterate** (*3) 1 = $[1, 3, 9, 27, 81, 243, \dots]$. Implemente a função **iterate** usando listas em compreensão.

5. (15%) Defina uma função `deleteNth`, que dado um inteiro n e uma lista l , remove elementos da lista l de n em n posições. Por exemplo: `deleteNth 3 "abcdefg" = "abdeg"` e `deleteNth 3 [1..10] = [1,2,4,5,7,8,10]`.

(a) Defina a função `deleteNth` recursivamente.

(b) Defina a função `deleteNth` usando listas em compreensão e/ou ordem superior.

6. (15%) Considere a seguinte declaração de tipo para árvores binárias:

`data Arv a = Folha | No a (Arv a) (Arv a)`

(a) Defina uma função `soma` que dada uma árvore de tipo numérico, calcule a soma dos valores na árvore.

(b) Defina uma função `somaArv` que, para duas árvores de tipo numérico t_1 e t_2 , calcula uma nova árvore cujos valores em cada nó são a soma dos valores de t_1 e t_2 nos nós respectivos. Assumimos que `Folha` é o elemento neutro da soma das árvores.

7. (10%) Responda (**apenas**) a uma das seguintes alíneas, usando indução matemática.

Nota: pode utilizar qualquer propriedade que tenha sido demonstrada nas aulas, ou demonstrar qualquer resultado adicional que facilite a prova.

- (a) Considerando as funções definidas na questão anterior, mostre que para qualquer árvore numérica t , $\text{soma} (\text{somaArv } t \ t) = 2 * \text{soma } t$.
- (b) Considerando as definições das funções `foldr`, `foldl` e `reverse` dadas nas aulas, assim como a função `flip` do prelúdio, definida como `flip f x y = f y x`, mostre que para quaisquer f, v e xs : $\text{foldl } f \ v \ xs = \text{foldr } (\text{flip } f) \ v \ (\text{reverse } xs)$.