

1. Escreva uma função  $f :: [\mathbf{Int}] \rightarrow [\mathbf{Int}]$  que retorna uma lista contendo todos os elementos da lista dada como argumento que ocorrem duas vezes em sucessão. Caso o elemento ocorra  $n$  vezes em sucessão ( $n \geq 2$ ), o elemento surgirá  $n - 1$  vezes em sucessão na lista dada como resultado. Não é necessário definir o valor para lista vazia.

Exemplos:  $f [1,2,2,3] = [2]$   $f [1,2,2,2,2,1] = [2,2,2]$   $f [3,2] = []$

- (a) Apresente uma solução que pode utilizar funções básicas e recursão, mas sem compreensão de lista
  - (b) Apresente uma solução com compreensão de lista, mas que não tenha recursão
2. Defina uma função  $g :: [\mathbf{Int}] \rightarrow \mathbf{Bool}$  que verifica que todo elemento de uma lista que está entre 0 e 100 (inclusive) é par. Utilize as funções **map**, **filter** e **foldr**.

Exemplos:  $g [1,26,153,72,68,9] = \mathbf{False}$

$g [12,153,74,10] = \mathbf{True}$

$g [] = \mathbf{True}$

$g [1,255] = \mathbf{False}$

3. Uma lâmpada é caracterizada por ser compacta ou incandescente. Além, disso, toda lâmpada possui o nome do seu fabricante e a potência como um valor em Watts.
  - (a) Defina o tipo algébrico **Lampada**, de acordo com as características descritas
  - (b) Estabeleça que exibir uma lâmpada resulta em uma string que começa com a palavra “Compacta”, no caso de lâmpada compacta, ou com a string “Incandescente”. Estas strings são seguidas do nome do fabricante e da potência da lâmpada. Ou seja, defina que o tipo **Lampada** é instância da classe **Show**.
  - (c) Estabeleça que o tipo **Lampada** é uma instância da classe **Eq**, de modo que duas lâmpadas são iguais se forem compactas e possuírem o mesmo fabricante e potência. O mesmo vale para lâmpadas incandescentes.