

Exercícios sobre Listas II

prof. André Rauber Du Bois

Universidade Federal de Pelotas
dubois@inf.ufpel.edu.br

1 Questionário

1. Defina a função

```
membro :: [Int] -> Int -> Bool
```

que retorna um booleano que diz se o inteiro esta na lista

2. Implemente a função

```
membroNum :: [Int] -> Int -> Int
```

que conta o número de vezes que o inteiro aparece na lista

3. Defina a função `membro` usando a função `membroNum`
4. Implemente a função

```
unico :: [Int] -> [Int]
```

que retorna uma lista com os números que aparecem apenas uma vez na lista argumento. Ex:

```
Hugs> unico [2,4,1,4,1,3]  
[2,3]
```

A função `memberNum` deve ser usada na definição de `unico`.

5. Se a lista argumento para `membro` está ordenada, não é necessário percorrer toda a lista para saber se o elemento está presente na lista. Implemente uma nova definição de `membro`, que use `iSort`.