

# 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/2` que recebe uma lista e um inteiro e que retorna um booleano dizendo se o inteiro está na lista.
2. Implemente a função `quantas_vezes/2` que recebe uma lista e um inteiro e conta quantas vezes o inteiro ocorre dentro da lista
3. Faça uma nova definição da função `membro/2` usando a função `quantas_vezes`
4. Implemente a função `unicos/1` que recebe uma lista como entrada e devolve uma lista contendo apenas os valores que ocorrem uma vez na lista. Ex:

```
iex(1)> unicos([2,4,1,4,1,3])  
[2,3]
```

A função `quantas_vezes` deve ser usada na definição de `unicos`.

5. Observe a implementação da função de ordenação `quick_sort`:

```
defmodule Lista2 do  
  def quick_sort([]), do: []  
  def quick_sort([h|t]) do  
    quick_sort(menores(h,t)) ++ [h] ++ quick_sort(maiores(h,t))  
  end  
  (...)   
end
```

A intuição dessa solução é a seguinte: deixar o primeiro elemento da lista (pivô) no meio, então colocar antes dele todos os elementos menores que o pivô ordenados e, depois dele, todos os elementos maiores que o pivô ordenados. O operador `(++)` serve para concatenar duas listas. Implemente as funções `menores` e `maiores` e veja o `quick_sort` funcionando!!!