



Faculdade de Computação  
Programação Funcional (BCC/BSI) - 1º Período  
**2a. Lista de Exercícios: Processamento de Listas em Haskell**

1. Mostre as listas geradas pelas seguintes expressões Haskell:

```
> [n*n | n<-[1..10], even n]
> [7 | n<-[1..5]]
> [(x,y) | x<-[1..5], y<-[4..7]]
> [(m,n) | m<-[1..3], n<-[1..m]]
> [j | i<-[1,-1,2,-2], i>0, j<-[1..i]]
> [a+b | (a,b) <- [(2,2), (3,3), (4,4), (5,5)]]
```

2. Defina uma função que dada uma lista de inteiros, retorna o número de elementos de valor superior a um número  $n$  qualquer.

```
> retornaSup 4 [3,2,5,6]
2
```

3. Defina uma função que dada uma lista de inteiros, retorna outra lista que contém apenas de elementos de valor superior a um número  $n$  qualquer.

```
> retornaListaSup 4 [3,2,5,6]
[5,6]
```

4. Escreva uma função que recebe duas listas de inteiros e produz uma lista de listas. Cada uma corresponde à multiplicação de um elemento da primeira lista por todos os elementos da segunda.

```
> mult_listas [1,2] [3,2,5]
[[3,2,5],[6,4,10]]
```

5. Escreva uma função para verificar se os elementos de uma lista são distintos.

6. Seja a função união abaixo, definida através da construção de listas por compreensão. Teste esta função e implemente a interseção utilizando a mesma estratégia.

```
uniao :: Eq t => [t] -> [t] -> [t]
uniao as bs = as ++ [b | b <- bs, not (pertence b as)]
```

7. Use a construção de listas por compreensão para definir uma função que recebe uma lista e retorna uma lista contendo apenas os elementos negativos da lista original,

8. Seja a função abaixo que recebe uma lista de pontos no plano cartesiano e calcula a distância de cada ponto à origem:

```
distancias :: [(Float,Float)] -> [Float]
distancias [] = []
distancias ((x,y):xys) = (sqrt (x^2 + y^2)) : (distancias xys)
```

Escreva uma outra versão da função distâncias, utilizando a construção de listas por compreensão.

9. Defina a função tabuada :: Int -> [(Int, Int, Int)] que dado um inteiro n produz uma lista da tabuada dos n.

```
> tabuada 5
[(5,1,5), (5,2,10), (5,3,15), ... , (5,10,50)]
```