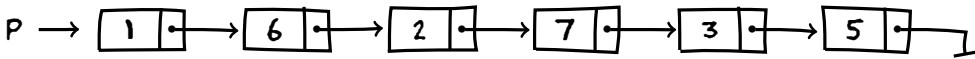


Contando os maiores que a média

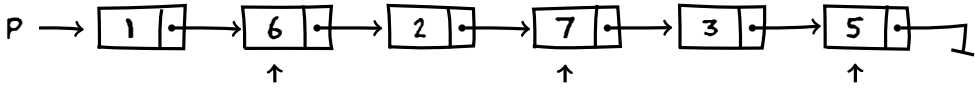
Imagine que o ponteiro `p` aponta para uma lista encadeada



A tarefa é

→ contar a quantidade de elementos maiores do que a média

No exemplo acima, isso nos daria



3 acima da média — (a média é 4.0)

Bom, a solução mais fácil consiste em utilizar a função que calcula a média.

E escrever uma função que conta os elementos maiores do que `k`.

A coisa fica assim

```
func CMqK-Rec (q,k,) // Conta Maiores que K
    Se ( q == Nulo )   Retorna (0) // começando a contar do 0
    resp = CMqK-Rec (q->prox,k)
    Se (q->num > k) // eu sou maior que o kara?
        resp = resp + 1
    Retorna (resp)
```

E a função `main()` fica assim

```
func main ()
    md = media-Rec (p,0,0)
    cont = CMqK (p,m)
    Imprime ('o número de elementos maiores do que a media é', cont)
```

Legal.

Mas essa não é a única maneira de fazer as coisas.

Quer dizer, a gente também pode modificar a função `media-Rec()` para que ela conte os elementos maiores do que a média na volta.

Para fazer isso, nós vamos utilizar dois valores de retorno.

A coisa fica assim

```
func CCMqM-Rec (q,soma,cont) // Calcula e Conta Maiores que a Média
    Se ( q == Nulo )   Retorna (soma/cont,0)
    Senão
        soma = soma + q->num // atualiza a soma
        cont = cont + 1 // atualiza o cont
        m,c = CCMqM-Rec (q->prox,soma,cont)
        Se (q->num > m)
            c++
        Retorna (m,c)
```

E a função `main()` fica assim

```
func main ()
    n,mm = CCMqM-Rec (p,0,0)
    Imprime ('o número de elementos maiores do que a media é', mm)
```