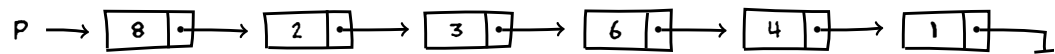


Contando o número de elementos

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



A tarefa é

→ *contar o número de elementos na lista*

Raciocinando de maneira recursiva, a gente resolve a tarefa assim

1. primeiro a gente conta o número de elementos no restante da lista
— *(fazendo uma chamada recursiva)*
2. e depois a gente soma 1 — *(para contar o 1o elemento)*

A coisa fica assim

```
func contaEl-Rec (q)

    Se ( q == Nulo )    Retorna (0)                // aqui não tem ninguém ...

    cont = contaEl-Rec (q->prox)                  // conta o restante da lista

    Retorna (cont + 1)                             // conta o primeiro elemento
```

Agora imagine que a tarefa é

→ *contar o número de elementos ímpares na lista*

Então, o raciocínio é quase a mesma coisa

1. primeiro a gente conta o número de elementos ímpares no restante da lista
— *(fazendo uma chamada recursiva)*
2. e depois a gente verifica se o primeiro elemento é ímpar
se ele for, a gente soma 1 no resultado

A coisa fica assim

```
func contaImpar-Rec (q)

    Se ( q == Nulo )    Retorna (0)                // aqui não tem ninguém ...

    cont = contaImpar-Rec (q->prox)                  // conta o restante da lista

    Se (q->num é ímpar)                                // verifica o primeiro elemento
        Retorna (cont + 1)

    Senão
        Retorna (cont)
```