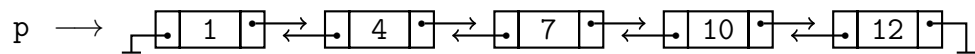


Imagine que o ponteiro **p** aponta para uma lista duplamente encadeada ordenada



A tarefa é

→ *inserir o número k na lista*

Bom, a novidade aqui é que nós não precisamos percorrer a lista com 2 ponteiros — (*e nem parar antes da hora ...*)

Quer dizer, nós percorremos a lista até encontrar o primeiro elemento maior do que **k**.

E fazemos a inserção atrás dele.

A única exceção é o caso em que o **k** entra na última posição da lista.

A coisa fica assim

```
func insereOrd (k, p)

    Se ( p == Nulo )                                     // inserção na lista vazia
        r = Novo (RegInt)
        r->num = k
        r->ant,prox = Nulo,Nulo
        Retorna(r)

    q = p
    Enquanto ( q->prox != Nulo )                         // procura o lugar do k
        Se ( q->num > k )    Quebra
        q = q->prox

    Se ( q->num > k )                                     // inserção atrás
        r = Novo (RegInt)
        r->num = k
        r->ant,prox = q->ant,q
        Se ( q->ant == Nulo )    p = r
        Senão                    (q->ant)->prox = r
        q->ant = r

    Senão                                                // inserção no fim
        r = Novo (RegInt)
        r->num = k
        r->ant,prox = q,Nulo
        q->prox = r

    Retorna (p)
```