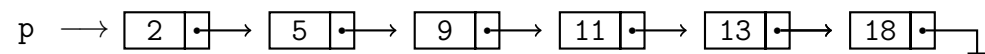


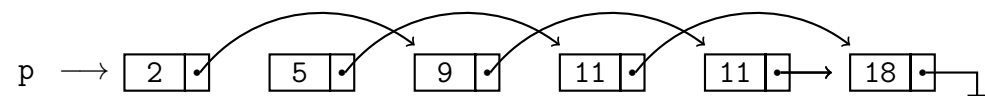
A lista vai-e-volta

Imagine que nós temos uma lista encadeada ordenada



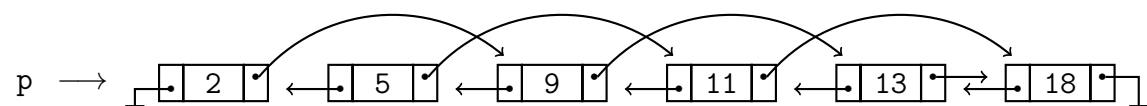
Para encontrar o número k a gente percorre a lista com o dedo — (*até encontrar ele, ou alguém maior do que ele*)

E a gente pode ir mais rápido pulando de 2 em 2



Mas assim, a gente pode passar do ponto.

Então é preciso ter um ponteiro para poder voltar



Essa é a *lista vai-e-volta*.

Aqui, a busca percorre a lista pulando de 2 em 2, até

- encontrar o número k
- encontrar alguém maior do que k
- ou chegar ao último elemento

Daí, caso o k não tenha sido encontrado, a gente dá um passo atrás — (*se houver alguém atrás ...*)

A coisa fica assim

```
func busca-VV (k, p)

    Se ( p == Nulo )    Retorna (Nulo)                // já acabei

    q = p

    Enquanto ( q->prox != Nulo )                      // vai até o fim
        Se ( q->num == k )    Retorna (q)
        Se ( q->num > k )     Quebra
        q = q->prox                // pulando de 2 em 2

    Se ( q->num == k )    Retorna (q)

    Senão
        Se ( q->ant != Nulo )    q = q->ant            // dá um passo atrás
        Se ( q->num == k )       Retorna (q)
        Senão                    Retorna (Nulo)
```