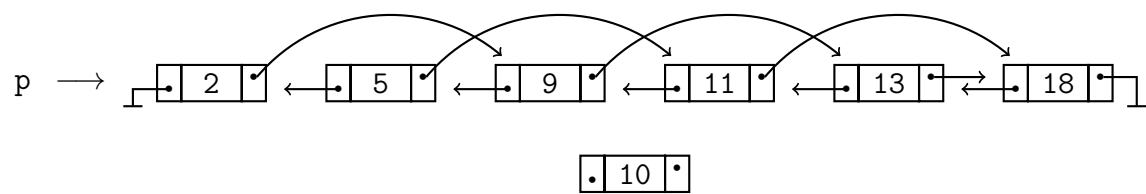


Inserção na lista vai-e-volta

Imagine agora que nós queremos inserir o 10 na nossa lista vai-e-volta

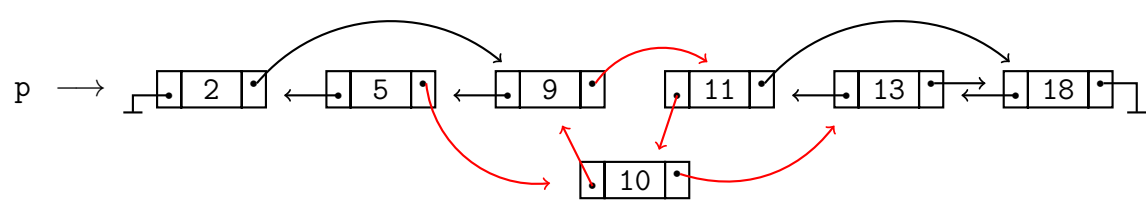


Então, andando para frente

- o 5 precisa apontar para o 10
- o 9 precisa apontar para o 11
- o 10 precisa apontar para o 13

e andando para trás

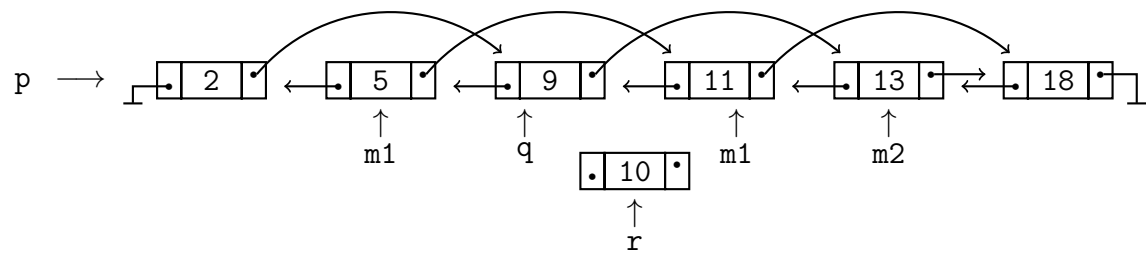
- o 11 precisa apontar para o 10
- o 10 precisa apontar para o 9



Para implementar esse procedimento, nós vamos utilizar uma função que retorna um ponteiro para o maior elemento da lista que é menor do que **k** — (ou *Nulo*, se esse elemento não existe)

Exercício: Implementar esse procedimento de busca

E daí, nós colocamos 3 mãozinhas



A coisa fica assim

```
func inserção-VV (k,p)

    r = Novo (RegInt )                                // cria o registro
    r->num = k

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

    q = buscaMmk-VV (k,p)

    Se ( q == Nulo )                                   // inserção na 1a posição
        r->ant = Nulo;   p->ant = r
        Se ( p->prox == Nulo )       r->prox = p           // p é o último
        Sse ( (p->prox)->ant == p )  r->prox = p->prox      // p é o penúltimo
        Senão                       r->prox = (p->prox)->ant
        Retorna ( r )

    Senão                                              // inserção no meio

        m1 = q->ant
        Se ( q->prox == Nulo )       m3 = Nulo;   m4 = Nulo
        Sse ( (q->prox)->ant == q )  m3 = q->prox;   m4 = q->prox
        Senão                       m4 = q->prox;   m3 = m4->ant

        r->prox = m3;   r->ant = q
        m1->prox = r;   q->prox = m2
        Se ( m2 != Nulo )
            q->prox = m2;   m2->ant = r
            Senão           q->prox =r
        Retorna ( p )
```