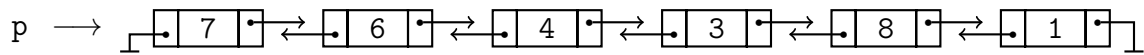


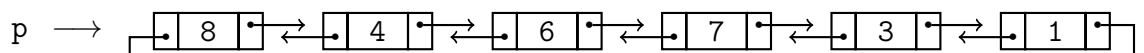
Imagine que o ponteiro p aponta para uma lista duplamente encadeada



A tarefa é

- Colocar todos os elementos pares no início da lista
e todos os elementos ímpares no final da lista
- (onde a ordem dos elementos pares e a ordem dos elementos ímpares não importa)

No exemplo acima, isso poderia nos dar



Bom, a ideia é simples.

Quer dizer, nós vamos percorrer a lista.

E toda vez que a gente encontrar um elemento par, nós vamos remover ele da lista e reinserir no início.

A coisa fica assim

```
func SPI (p)                                     // Separando Pares e Ímpares

    Se ( p == Nulo OU p->prox == Nulo )          // nada a fazer ...
        Retorna (p)

    q = p
    Enquanto ( q != Nulo )
        Se (q->num é par E q->ant != Nulo)
            m1 = q->ant;   m2 = q   m3 = q->prox
            m1->prox = m3

            Se (m3 != Nulo)   m3->ant = m1
            m2->ant,prox = Nulo,p

            p = m2
            q = m3

        Senão
            q = q->prox

    Retorna (p)
```