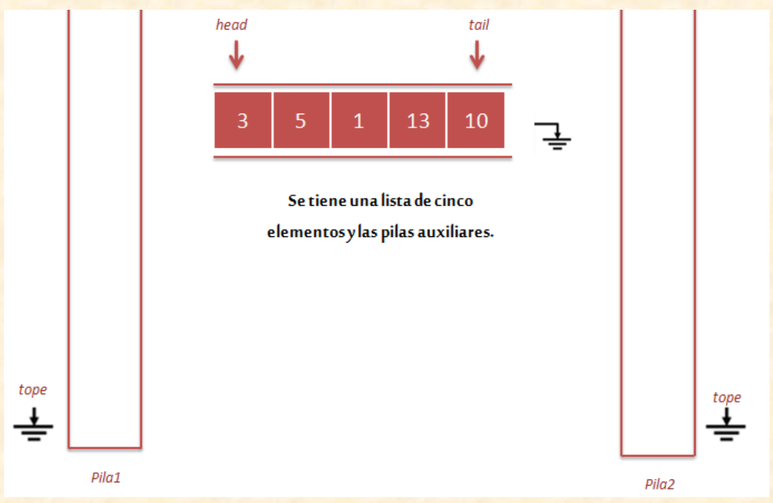
Método de ordenamiento por pilas

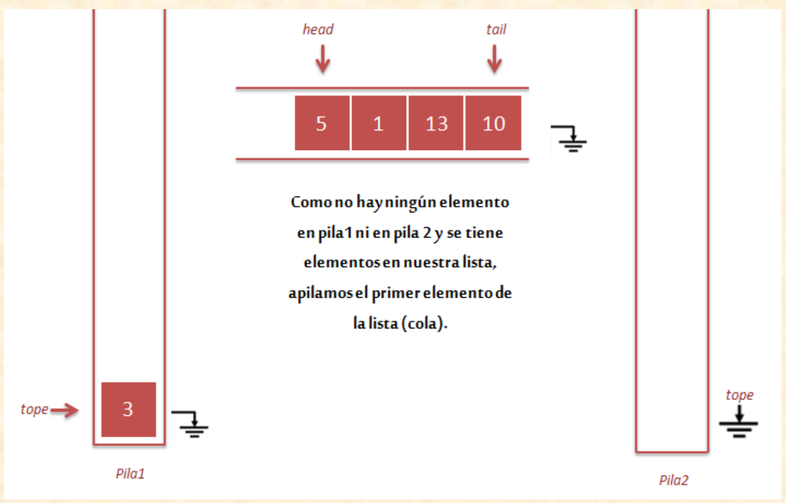
¿En qué consiste?

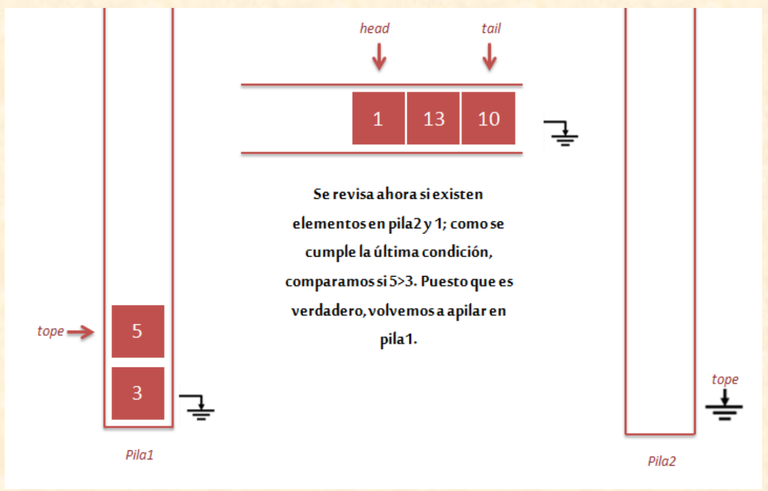
Mediante dos **pilas auxiliares, se va ordenando cada elemento de la lista al ir comparando los topes de ambas con el primer elemento de la lista.**
Durante el proceso de ordenamiento, la lista se comporta como una **cola.**

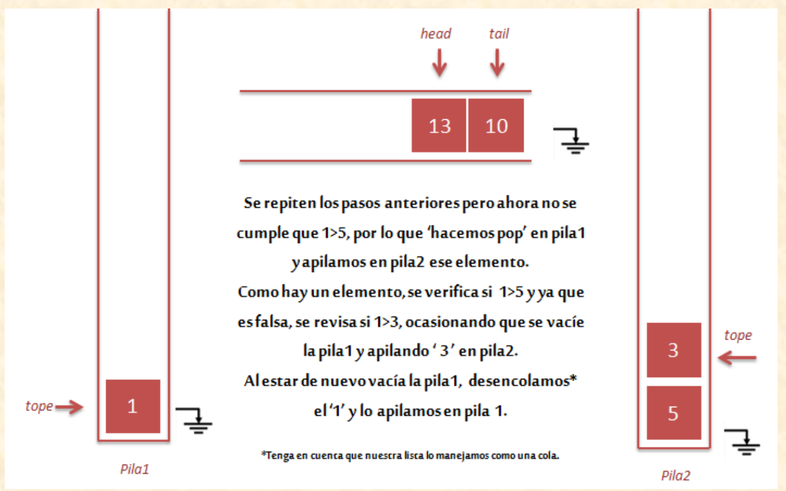
Funciones necesarias

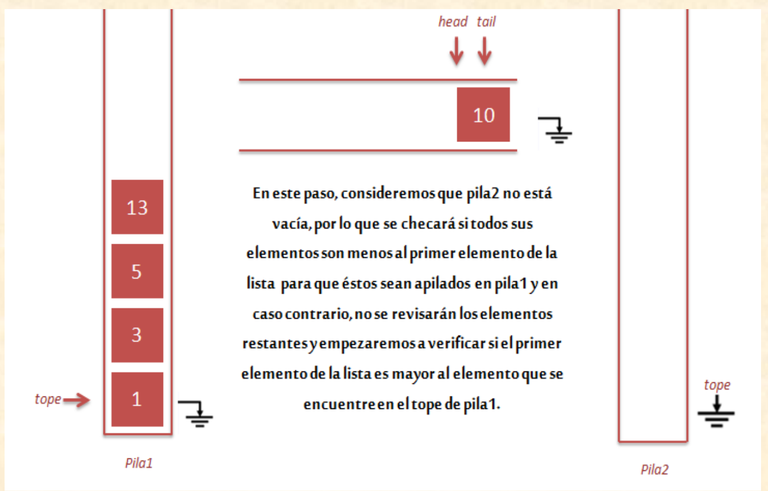
- SacarP (desencolar)
- Push
- Pop
- leerUltimo (top)
- leerPrimero(peek)

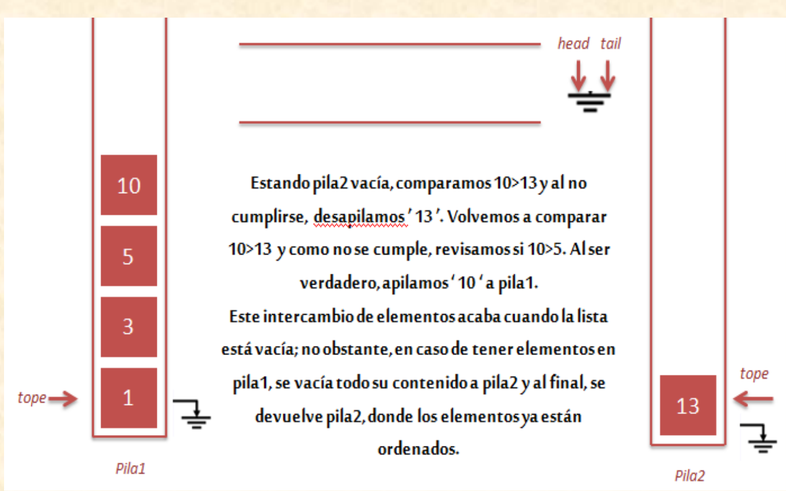


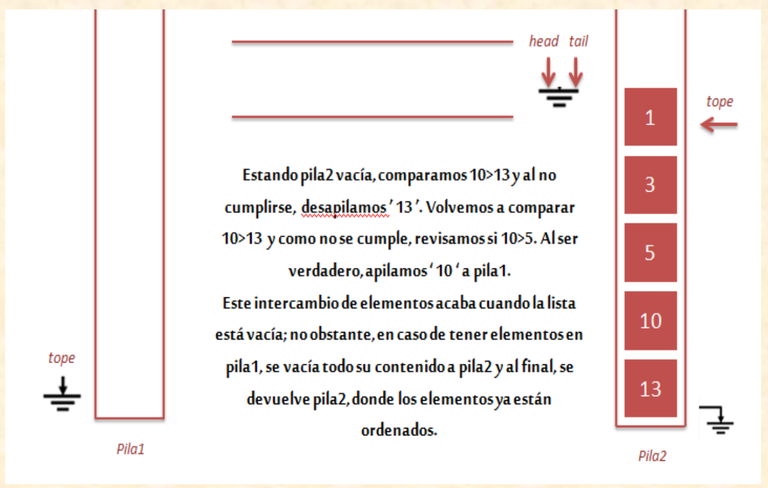












Como podemos notar, lo que se regresa es una pila e inicialmente lo que se mandó es una lista por lo que causa una incompatibilidad. Esto no es problema si todo lo trabajamos desde un inicio con listas. Sin embargo, deben utilizarse como pilas y la lista a ordenar como cola.

Al realizar esa modificación, se debe mandar al final todos los elementos a la pila1 (en caso de querer una lista ordenada de menor a mayor).

Pseudocódigo

```
OrdenamientoInterno(A[]: Entero): DEV L0[]:ENTERO
Pila1 = ∅
Pila2 = ∅
SI A = ∅:
    DEV A
FIN SI
Apilar(pila1,desencolar(A))
MIENTRAS A ≠ ∅:
    SI pila1≠ ∅:
        Apilar(pila1,desencolar(A))
    FIN SI
    Si A = ∅:
        RomperCiclo
    FIN SI
    MIENTRAS pila2≠ ∅:
        Si A[0] > pila2[length(pila2)-1]:
            Apilar(pila1,pop(pila2))
        FIN SI
        EN CASO CONTRARIO:
            RomperCiclo
        FIN EN CASO CONTRARIO
    FIN MIENTRAS
    SI A[0]> pila1[length(pila1)-1]:
        Apilar(pila1,desencolar(A))
    FIN SI
    EN CASO CONTRARIO:
        Apilar(pila2,pop(pila1))
    FIN EN CASO CONTRARIO
FIN MIENTRAS
SI pila1≠ ∅:
    MIENTRAS pila1≠ ∅:
        Apilar(pila2,pop(pila1))
    FIN MIENTRAS
FIN SI
DEV pila2
```

Algoritmo de ordenamiento creado por Suxo Pacheco Elsa Guadalupe· Facultad de Ingeniería· Semestre 2019-1