

Video explicativo

### CONSIGNA

Escribir una función `void sortchild(tree<int> &T, tree<int> &Q);` que crea una copia Q de T pero donde cada lista de hijos está ordenada de menor a mayor. Si hay varios nodos en la lista con el mismo valor deben quedar en el orden en el que estaba (estabilidad).

### Ejemplo:

T=(1 3 2 (4 6 5))	=>	Q=(1 2 3 (4 5 6))
T=(0 (3 4 3) (1 2) 1 (4 3 1))	=>	Q=(0 (1 2) 1 (3 3 4)
(4 1 3))		
T=(1 (4 4 (1 (2 4 1))) (2 4 (4 1)))	=>	Q=(1 (2 4 (4 1)) (4 (1
(2 1 4)) 4))		
T=(0 (2 3 2 1 (1 1)) (0 (1 0) 4 (3 3)))	=>	Q=(0 (0 (1 0) (3 3) 4)
(2 1 (1 1) 2 3))		

### AYUDA

- En el wrapper si el árbol está vacío no hay que hacer nada.
- Caso contrario copiar la raíz de T en la raíz de Q.
- Escribir una función recursiva que toma como argumentos los árboles T, Q e iteradores sobre los mismos nt, nq.
- Ir recorriendo los hijos ct de nt y de acuerdo a su valor ubicar la posición cq de los hijos de nq donde corresponde que vaya el valor para que quede ordenado.
- Insertar un nuevo nodo en cq y aplicar recursivamente la función sobre ct, cq.