VÉRIFICATION DE L'ALGORITHME

- Initialisation
  - ✓ k=p, donc T[p…k-1] est vide avec les k-p plus petits éléments de L et R. i=j=1 donc L[i]et R[j] sont bien les plus petits éléments de leurs tableaux respectifs.
- Conservation
  - ✓ Si L[i] inférieur à R[j], alors L[i] est le plus petit élément non copié. T[p…k-1] contient les k-p plus petits éléments, après la copie de L[i], T[p…k] contient bien les k-p+1 plus petits éléments. L'incrémentation de k et i recrée l'invariant sans perturber cette vérité. Si L[i] > R[j], on a l'action idoine.
- Terminaison
  - ✓ k=r+1. T[p…k-1] ou T[p…r] contient les k-p=r-p+1 plus petits éléments de L[1…Sl+1] et R[1…Sr+1] dans l'ordre croissant. Seules les sentinelles n'ont pas été copiées car à eux deux la taille est r-p+3=Sl+Sr+2 Donc tout est trié et fusionné correctement.

COMPLEXITÉ

fusion(T,p,q,r)

$S_l \leftarrow q - p + 1$
$S_r \leftarrow r - q$
création des tableaux $L[1 \ldots S_l + 1] et R[1 \ldots S_r + 1]$
pour i allant de 1 à S_l
    faire $L[i] \leftarrow T[p + i - 1]$
pour i allant de 1 à S_r
    faire $R[i] \leftarrow T[q + i]$
$L[S_l + 1] \leftarrow fin$
$R[S_r + 1] \leftarrow fin$
$i \leftarrow 1$
$j \leftarrow 1$
pour k allant de p à r
    faire si $L[i] \leq R[j]$
        alors $T[k] \leftarrow L[i]$
            $i \leftarrow i + 1$
        sinon $T[k] \leftarrow R[j]$
            $j \leftarrow j + 1$

coût	fois
c_1	1
c_2	1
c_3	1
c_4	n_l
c_5	$n_l - 1$
c_6	n_r
c_7	$n_r - 1$
c_8	1
c_9	1
c_{10}	1
c_{11}	1
c_{12}	n
c_{13}	n - 1
c_{14}	n_l
c_{15}	n_l
c_{16}	n_r
c_{17}	n_r

O(n)

O(n)