

ALGORITHMES DE TRI

**Master : Cyber Sécurité et Intelligence Artificielle
CSIA**

Année universitaire : 2025/ 2026

TRI RAPIDE

- Le principe est de choisir une valeur dans le tableau appelée pivot (par exemple la première valeur du tableau) et de déplacer avant elle toutes celles qui lui sont inférieures et après elle toutes celles qui lui sont supérieures. Réitérer le procédé avec la tranche de tableau inférieure et la tranche de tableau supérieure à ce pivot.

20, 15, 10, 35, 19, 13, 5, 3, 12, 7, 16, 40, 25, 38

15, 10, 19, 13, 5, 3, 12, 7, 16, 20, 35, 40, 25, 38
à trier ! à trier !

TRI RAPIDE

- **DIVISER:** Diviser le tableau en deux tableaux selon le pivot choisi : $T1[\text{debut}..\text{pivot}-1]$ et $T2[\text{pivot}+1..\text{fin}]$
- **REGNER:** trier (par trie rapide) les deux tableaux
- **COMBINER:** combiner les 2 tableaux: $T[\text{debut}..\text{fin}] = T1[\text{debut}..\text{pivot}-1] + T2[\text{pivot}+1..\text{fin}]$ est trié

TRI RAPIDE

Tri_Rapide (T: tableau, debut, fin : entier)

Debut

SI debut < fin alors

Pivot ← partitionner(T, debut, fin , pivot)

Tri_Rapide(T, debut, pivot-1)

Tri_Rapide(T, pivot+1, fin)

FSI

Fin

TRI RAPIDE

Partitionner (T: tableau, debut, fin, pivot: entier): entier

Debut

permuter (T[pivot], T[fin])

$J \leftarrow$ premier

Pour $i \leftarrow 1$ à fin -1 faire

 SI $T[i] \leq T[fin]$ alors

 Permuter (T[i], T[j])

$j \leftarrow j + 1$

 FSI

FP

Permuter (T[fin] , T[j])

Partitionner $\leftarrow j$

Fin

TRI RAPIDE

complexité

Partitionner (T: tableau, debut, fin, pivot: entier): entier

Debut

permuter (T[pivot], T[fin])

$J \leftarrow$ premier

Pour $i \leftarrow 1$ à fin -1 faire

 Si $T[i] \leq T[fin]$ alors

Pire de cas: n-1 fois

 Permuter (T[i], T[j])

$j \leftarrow j + 1$

 FSI

FP

Permuter (T[fin] , T[j])

Partitionner $\leftarrow j$

Fin

$$T_{\text{part}}(n) = c1 * n + c2 \rightarrow O(T_{\text{part}}) = O(n)$$

TRI RAPIDE

complexité

Tri_Rapide (T: tableau, debut, fin, pivot : entier) $T(n)$

Debut

Si debut < fin alors

Pivot \leftarrow partitionner(T, debut, fin , pivot) $T_{part}(n)$

Tri_Rapide(T, debut, pivot-1) $T(\text{pivot-debut}+1)$

Tri_Rapide(T, pivot+1, fin) $T(\text{fin-pivot})$

FSI

Fin

$$T(n) = T(\text{pivot-debut}+1) + T(\text{fin-pivot}) + T_{part}(n)$$

TRI RAPIDE

choix du pivot et complexité

- **Cas 1: Pivot aléatoire.** Le pire cas intervient quand le partitionnement produit une région à $n-1$ éléments et une région à un élément: $T(n) = T(n-1) + T(1) + T_{\text{part}}(n) = T(n-1) + f(n) \rightarrow O(T) = O(n^2)$
- **Cas 2: Pivot Arbitraire** (premier élément, dernier élément ou élément en milieu). *Même chose que le cas 1.*
- **Cas 3: Pivot optimal** comme la valeur médiane qui permet de couper le tableau en deux parties égales de taille $n/2$:
 $T(n) = 2T(n/2) + T_{\text{part}}(n) \rightarrow O(T) = O(n \log_2 n)$