```
TRI_FUSION(T,p,r)
  Si p<r
    q ← ⌊(p+r-1)/2⌋
    Tri_fusion(T,p,q)
    Tri_fusion(T,q+1,r)
    Fusion(T,p,q,r)

FUSION(T,p,q,r)
  G tableau de taille q-p+1.
  Pour i = 1 à q-p
    G[i] ← T[p+i-1]
  G[q-p+1] ← +∞
  D tableau de taille r-q
  Pour i = 1 à r-q-1
    D[i] ← T[q+i]
  D[r-q] ← +∞
  i ← 1
  j ← 1
  Pour k = p à r
    Si L[i] ≤ R[j]
      T[k] ← L[i]
      i ← i+1
    Sinon
      T[k] ← R[j]
      j ← j+1.
```

```
TRI_RAPIDE(T,p,r)
  Si p<r
    q = Partition(T,p,r)
    Tri_rapide(T,p,q-1)
    Tri_rapide(T,q+1,r)

PARTITION(T,p,r)
  x ← T[r]
  i ← p
  Pour j = p à r-1
    Si T[j] ≤ x
      T[i] ↔ T[j]
      i ← i+1
  T[i] ↔ T[r]
  Renvoyer i
```

TF : Complexité = nbre de comparaisons (pire cas)

→ n comparaisons sur un tableau de taille n

$$c(n) = c\left(\left\lfloor \frac{n}{2} \right\rfloor\right) + c\left(\left\lceil \frac{n}{2} \right\rceil\right) + n$$

et $c(0) = c(1) = 0$.

On admet que la complexité est croissante :

→ on résout pour $n = 2^{k+1}$

$$\Rightarrow c(2^{k+1}) = c(2^k) + c(2^k) + 2^{k+1}$$

$$\Rightarrow \frac{c(2^{k+1})}{2^{k+1}} = \frac{c(2^k)}{2^k} + 1$$

$$\Rightarrow \frac{c(2^k)}{2^k} = k \Rightarrow c(n) = \Theta(n \log_2 n)$$

TR : Complexité pire cas :

→ Partition = n comparaisons sur un tableau de taille n

$C_p(n+1) = n + C_p(n) \Rightarrow C_p(n) = \Theta(n^2)$

(l'un des deux tableaux est vide).

TR : Complexité moyenne :

Pivot en position j ⇒ tableaux de taille $j-1$ et $n-j$

Proba : toutes les permutations sont équiprobables.

⇒ Pivot en position j → proba $\frac{1}{n}$.

$$C_{moy}(n) = n - 1 + \frac{1}{n}\sum_{j=1}^{n}\left(C_{moy}(j-1) + C_{moy}(n-j)\right) \text{ et } C_{moy}(0) = C_{moy} = 0.$$

$$= n - 1 + \frac{2}{n}\sum_{j=1}^{n-1} C_{moy}(j)$$

$$\Rightarrow n\left(C_{moy}(n) - n + 1\right) = 2\sum_{j=1}^{n-1} C_{moy}(j)$$

$$\text{et } (n-1)\left(C_{moy}(n-1) - n + 2\right) = 2\sum_{j=1}^{n-2} C_{moy}(j)$$

$$\Rightarrow n C_{moy}(n) - (n-1)C_{moy}(n-1) - 2n + 2 = 2C_{moy}(n-1)$$

$$\Rightarrow \frac{C_{moy}(n)}{n+1} = \frac{C_{moy}(n-1)}{n} + \frac{2(n-1)}{n(n+1)} \leq \frac{C_{moy}(n-1)}{n} + \frac{2}{n+1}.$$

$$\Rightarrow C_{moy}(n) \leq (n+1)\sum_{k=1}^{n}\frac{2}{k+1} = 2(n+1)(H_{n+1} - 1) \underset{n\to+\infty}{\sim} 2n \ln n$$

$$\Rightarrow C_{moy}(n) = \Theta(n \log_2 n).$$

Constantes : Fusion ≈ 1 et Rapide ≈ 1,38.

Mais affectation de variables dans tri fusion alors que tri rapide est en place.

⇒ En pratique, tri rapide est plus rapide que le tri fusion.

DVT3 : Algorithme de Dijkstra : Complexité (Ref : Cormen)

Donnée : $G=(V,E)$ un graphe $W: E \to \mathbb{N}$ fonction de poids, $s \in V$ une source

But : Calculer $d(s,v)$ pour tout $v \in V$, et donner un chemin de poids minimal.

On note $V=\{1,\dots,n\}$, on va construire les tableaux d de distance et π de prédécesseurs à l'aide d'une file de priorité min F, dont les clés sont les valeurs de d.

On va utiliser les opérations :

- INSERER(F, v, clé) : insère le sommet v avec sa clé $d[v]$ dans F
- EXTRAIRE_MIN(F) : renvoie le sommet $v \in F$ tel que $d[v]$ est minimal dans F. v est enlevé de F.
- DIMINUER_CLE(F, v, clé) : si $v \in F$, on met $d[v] \leftarrow$ clé, puis réorganise F.

RELACHER(u, v, W) — $R(k)$
- Si $d[v] > d[u] + W(u,v)$ → 1
  - DIMINUER_CLE(F, v, $d[u]+W(u,v)$) → $D(n-k-1)$
  - $\pi[v] \leftarrow u$

DIJKSTRA(G, W, s) — $C(n)$
- Initialiser π à NIL → n
- Créer le tableau d → 1
- $F = \emptyset$ → 1
- Pour $v \in V$, → n
  - INSERER(F, v, ∞) → $I(k)$ (← en vrai ici insertion linéaire (car clé = $+\infty$))
- DIMINUER_CLE(F, s, 0) → $D(n)$
- Tant que $F \neq \emptyset$ → n
  - $u \leftarrow$ EXTRAIRE_MIN(F) → $E(n)$
  - Pour $v \in \mathrm{Adj}(u)$ si $v \in F$ → $|\mathrm{Adj}(u)|$
    - RELACHER(u, v, W) → $R(k)$ (sortir directement ici ?)

Complexité : (pire cas)

On note $I(n)$, $E(n)$, $D(n)$ les complexités de INSERER, EXTRAIRE_MIN, DIMINUER_CLE

$$\Rightarrow C(n) = 2n + 1 + \sum_{k=0}^{n-1} I(k) + \sum_{k=1}^{n} \left(E(k) + \mathrm{Adj}\,(2 + D(k))\right) \leq n I(n) + n E(n) + 2A(2 + D(n)) + O(n$$

F	$I(n)$	$E(n)$	$D(n)$	Matrice d'adjacence	Liste d'adjacence
Tableau	1	n	1	$n + n^2 + n^2 \times 3 = \Theta(n^2)$	$\Theta(n^2 + A) = \Theta(n^2)$
Tas min	$\log n$	$\log n$	$\log n$	$n \log n + n \log n + n^2(2 + \log n) = \Theta(n^2 \log n)$	$\Theta((n + A) \log n)$

taille de la + gde liste d'adj.

⇒ liste d'adjacence tout le temps

⇒ graphe peu dense : tas min — tas min mieux

⇒ graphe dense : tableau