

Fiche de révision 12 :

Algorithmes de tris

Thème 7 ♦

Tri Sélection

Principe : rechercher le plus petit élément et le placer en début de liste puis recommencer avec le reste de la liste à trier.

Algorithme :

Fonction **minimum**(tableau t de longueur n)
indiceMin ← 0

Pour i allant de 0 à n-1

 Si $t[i] < t[\text{indiceMin}]$, alors :

 indiceMin ← i

 Fin Si

Fin Pour

renvoyer indiceMin

Fonction **triSelection**(tableau t de longueur n)

pour i allant de 0 à n-2

 indiceMin ← minimum(t[i:n-1])+i

 Si indiceMin ≠ i, alors :

 Echanger t[i] et t[indiceMin]

 Fin si

Fin Pour

renvoyer t

Tri Insertion

Principe : insérer un élément à la bonne place dans la partie déjà triée de la liste

Algorithme :

def **triInsertion**(tableau t de longueur n):

Pour i allant de 1 à n-1

temp ← t[i]

j ← i-1

Tant que j > 0 et t[j] > temp :

t[j+1] ← t[j]

j ← j-1

Fin Tant que

t[j] ← temp

Fin Pour

renvoyer t

Programme (Python) :

```
def minimum(t):
    indiceMin=0
    for i in range (0,len(t)-1):
        if t[i]<t[indiceMin]:
            indiceMin=i
    return indiceMin
def triSelection(t):
    for i in range(0,len(t)):
        indicemin=minimum(t[i:])+i
        if t[i]!=t[indicemin]:
            t[i],t[indicemin]=t[indicemin],t[i]
    return t
```

Programme (Python) :

```
def triInsertion(t):
    for i in range(1,len(t)):
        temp=t[i]
        j=i-1
        while j>0 and t[j]>temp:
            t[j+1]=t[j]
```

```
    j-=1
    t[j]=temp
return t
```