

A retenir du chapitre 10 :

Algorithmes de tri 1



- On dit qu'un tri est :
 - **comparatif** s'il utilise des comparaisons (inégalités) entre les éléments de la liste à trier,
 - **stable** s'il conserve l'ordre initial des éléments qui sont égaux,
 - **en place** s'il ne crée pas de copie des données à trier.

- Algorithme de tri à bulles :

```
def TriBulles(l) :  
    n=len(l)  
    for i in range(n-1) :  
        for j in range(0,n-i-1) :  
            if l[j]>l[j+1] :  
                l[j],l[j+1]=l[j+1],l[j]  
    return l
```

- Algorithme de tri par sélection :

```
def TriSelection(l) :  
    n=len(l)  
    for i in range(n) :  
        imin=i #recherche du minimum à partir du rang i  
        m=l[i]  
        for j in range(i+1,n) :  
            if l[j]<m :  
                imin=j  
                m=l[j]  
        l[imin],l[i]=l[i],l[imin] # on place le minimum en position i  
    return l
```

- Algorithme de tri par insertion :

```
def TriInsertion(l) :  
    n=len(l)  
    for i in range(n) :  
        k=i  
        cle=l[i]  
        while k>0 and cle<l[k-1] :  
            l[k]=l[k-1] # décalage vers la droite des valeurs supérieures à la clé  
            k-=1  
        l[k]=cle# insertion de la clé  
    return l
```