

Exercice 1.

1.1) Tri à bulle

```
def tri_Bulle(T):
    N = len(T)
    changement = True
    while changement == True:
        changement = False
        for k in range(N-1):
            if T[k] > T[k+1]:
                T[k], T[k+1] = T[k+1], T[k]
                changement = True
        N = N-1
    return T
```

1.2) Tri à bulle décroissant

```
def tri_Bulle(T):
    N = len(T)
    changement = True
    while changement == True:
        changement = False
        for k in range(N-1):
            if T[k] < T[k+1]:
                T[k], T[k+1] = T[k+1], T[k]
                changement = True
        N = N-1
    return T
```

Exercice 2.

2.1) Tri par insertion

```
def tri_Insertion(T): # T est le tableau à trier
    N = len(T) # N = longueur du tableau
    # Balayage du tableau du :
    for i in range(1,N):
        x = T[i] # élément à insérer
        k = i # k est son indice
        # Insertion dans la partie triée :
        while k > 0 and T[k-1] > x:
```

Décalage sur la gauche :

T[k] = T[k-1]

k = k-1

T[k] = x

return T # Si on retourne le résultat

2.2) Tri par insertion décroissant

```
def tri_Insertion(T): # T est le tableau à trier
    N = len(T) # N = longueur du tableau
    # Balayage du tableau du :
    for i in range(1,N):
        x = T[i] # élément à insérer
        k = i # k est son indice
        # Insertion dans la partie triée :
        while k > 0 and T[k-1] < x:
            # Décalage sur la gauche :
            T[k] = T[k-1]
            k = k-1
        T[k] = x
    return T # Si on retourne le résultat
```

Exercice 3.

3.1) Recherche de valeur maximale

```
def maximum(L):
    m = L[0]
    for k in range(1,len(L)):
        if L[k] > m:
            m = L[k]
    return m
```

3.1) Recherche de l'indice d'une valeur maximale

```
def indiceMaximum(L):
    ind = 0
    for k in range(1,len(L)):
        if L[k] > L[ind]:
            ind = k
    return ind
```

Exercice 4.

```
# 4.1) Fonction indiceMax(T, i)
def indiceMax(T, i):
    iMax = i
    for k in range(i):
        if T[iMax] < T[k]:
            iMax = k
    return iMax

# 4.2) Fonction tri_selection

def tri_selection(T):
    N = len(T)
    for i in range(N-1, 0, -1):
        imax = indiceMaximum(T, i)
        T[imax], T[i] = T[i], T[imax]
```

```
return (T1[(n-1)//2]+T1[n//2]) / 2
```

Exercice 5.

```
def triComptage(L):
    nb0, nb1 = 0, 0
    for k in range(len(L)):
        if L[k] == 0:
            nb0 += 1
        elif L[k] == 1:
            nb1 += 1
    Lc = []
    for k in range(nb0):
        Lc.append(0)
    for k in range(nb1):
        Lc.append(1)
    return Lc
```

Exercice 6.

```
def mediane(T):
    T1 = T[:]
    tri_Insertion(T1)
    n = len(T1)
    if n%2 == 1:
        return T1[n//2]
    else:
```