

Préparation à l'épreuve pratique NSI

05

Tri par insertion d'une table de valeurs

Nom :**Note : / 20****Classe :**

1 Spécification du programme à réaliser

1.1 Description générale

On considère un tableau t [] de n éléments numérotés de 0 à $n-1$. L'objectif consiste à trier le tableau en utilisant l'algorithme de tri par insertion.

<u>Entrées</u>	<u>Sorties</u>
Le tableau t [] de n éléments.	Le tableau trié.

1.2 Pseudo code de l'algorithme

```
pour i de 1 à n - 1
    # mémoriser T[ i ] dans x
    x ← T[ i ]
    # décaler vers la droite les éléments T[ 0 ] ..T[ i - 1 ]
    # qui sont plus grands que x en partant de T[ i - 1 ]
    j ← i
    tant que j > 0 et T[ j - 1 ] > x
        T[ j ] ← T[ j - 1 ]
        j ← j - 1
    fin tant que
    # placer x dans le "trou" laissé par le décalage
    T[ j ] ← x
fin pour
```

1.3 Amélioration

- On traitera le cas où la liste donnée en paramètre ne contient aucun élément. La fonction retournera la valeur **None** dans ce cas.

2 Codage et mise en œuvre

2.1 Script de l'exercice



 NSI-PROG-005-Tri-Par-Insertion.py

2.2 Fonction à compléter

Votre fonction à réaliser

```
def tri_par_insertion(liste):  
    '''  
    Réalise le tri par insertion de la liste passée  
    en paramètre  
    '''  
  
    return liste
```

2.3 Résultats attendus

```
>>> (executing lines 1 to 88 of "ALGO_Tri-Par-Insertion-corrige.py")  
Liste de départ : [56, 10, -2, 52, -8, 41, -13]  
Liste triée      : [-13, -8, -2, 10, 41, 52, 56]  
  
Liste de départ : [250, -10, 8, 52, 84]  
Liste triée      : [-10, 8, 52, 84, 250]  
  
Liste de départ : [-22, -59, 73, 84, 66, 53, 86, 27, -84, 42, 54, -47]  
Liste triée      : [-84, -59, -47, -22, 27, 42, 53, 54, 66, 73, 84, 86]  
  
Liste de départ : []  
Liste triée      : None  
  
Liste de départ : [4]  
Liste triée      : [4]
```

