

Les calculatrices sont interdites. La qualité de la rédaction, la clarté et la précision des raisonnements seront prises en compte de manière importante dans l'appréciation des copies.

Exercice. L'objectif de l'exercice est d'étudier l'algorithme du tri par insertion, déjà rencontré en TP. Pour rappel, son fonctionnement est le suivant :

- On prend en entrée une liste d'entiers $L = [a_0, a_1, \dots, a_{n-1}]$ de longueur n .
- On note $L_0 = [a_0]$. C'est une liste triée, puisqu'elle ne comporte qu'un seul élément.
- Pour tout $k \in \llbracket 1, n-1 \rrbracket$, on s'intéresse à la liste préalablement triée $L_{k-1} = [b_0, b_1, \dots, b_{k-1}]$. On y insère l'élément a_k en procédant par comparaison puis éventuels décalages vers la gauche. À la fin de l'opération, on obtient une liste $L_k = [b_0, b_1, \dots, b_k]$ triée.

1. On s'intéresse à la liste $L = [5, 2, 4, 6, 1, 3]$. Expliciter le contenu des listes L_0, L_1, L_2, L_3, L_4 et L_5 fournies par les étapes successives de l'algorithme.

Pour le reste de l'étude, on se basera sur l'implémentation en Python suivante, qui trie sur place la liste d'entiers L :

```
def TriInsertion(L):
    for k in range(1, len(L)):
        c=L[k]
        j=k
        while j>0 and c<L[j-1]:
            L[j]=L[j-1]
            j-=1
        L[j]=c
```

2. (a) Proposer une précondition et une postcondition pour cet algorithme.
 (b) Montrer que l'algorithme termine.
 (c) Déterminer un invariant de boucle pour la boucle **for**.
 (d) En déduire que l'algorithme est correct.
3. Soit $n \in \mathbb{N}^*$ la taille de la liste L , on note $T(n)$ le nombre de comparaisons effectuées dans le pire des cas par cet algorithme de tri.
 (a) Déterminer le nombre de comparaisons lors d'un passage dans la boucle **while**.
 (b) Soit $k \in \llbracket 1, n-1 \rrbracket$. En partant de la valeur $j = k$, déterminer le nombre de comparaisons effectuées dans le pire des cas lors de l'exécution complète de la boucle **while**.
 (c) Déterminer l'expression de $T(n)$ en fonction de n , sous la forme la plus simplifiée possible.
 (d) Donner une estimation asymptotique de $T(n)$.
4. Soit $n \in \mathbb{N}^*$ la taille de la liste L , on note maintenant $A(n)$ le nombre de comparaisons effectuées dans le meilleur des cas par l'algorithme de tri. Déterminer l'expression de $A(n)$ en fonction de n , puis une estimation asymptotique de $A(n)$.