

Nom: Prénom: **Exercice 1 : Complexité des algorithmes**

1. Si un algorithme a une complexité de $O(n)$, que se passe-t-il pour le temps d'exécution si la taille de la liste est multipliée par 10?
2. Si un algorithme a une complexité de $O(n^2)$, que se passe-t-il pour le temps d'exécution si la taille de la liste est multipliée par 10?
3. Si un algorithme a une complexité de $O(\log n)$, que se passe-t-il pour le temps d'exécution si la taille de la liste est multipliée par 64?

Exercice 2 : On considère une liste triée L.

1. Ecrire un algorithme de recherche dichotomique pour trouver la position d'un élément x dans la liste. Si l'élément n'est pas présent, l'algorithme doit retourner -1.
2. Indiquer quel est le variant de boucle de cet algorithme.
3. Indiquer quel est l'invariant de boucle de cet algorithme.
4. Quelle est la complexité de cet algorithme?

Exercice 3 : On considère le code suivant :

1. Quel est le variant de boucle?
2. Justifier que la boucle termine.
3. Quelle est la complexité de cet algorithme?
4. Qu'affiche ce code si $L = [10, 20, 30, 40, 50]$?

Code Python

```

1 i = 0
2 n = len(L)
3 while i < n:
4     print(L[i])
5     i += 2

```

Exercice 4 : Débogage

1. Le code suivant pour la recherche naïve contient une erreur classique. Laquelle?
2. Corriger le code pour qu'il fonctionne correctement.

Exercice 5 : Débogage

Le code suivant présente une erreur, l'identifier et la corriger.