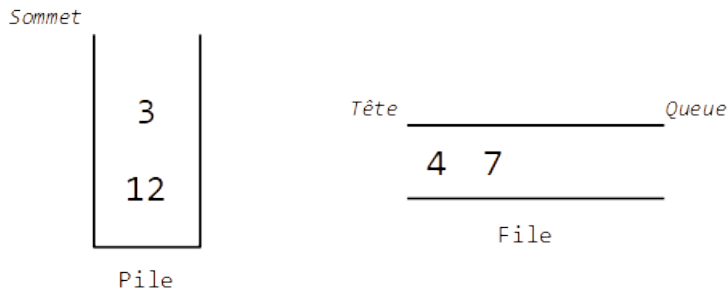
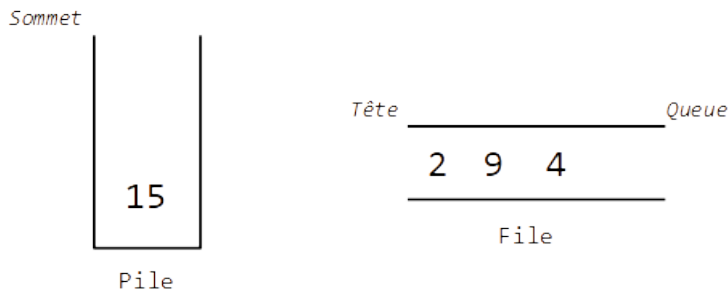


Exercice 1 - Structures abstraites de données et programmation en général.

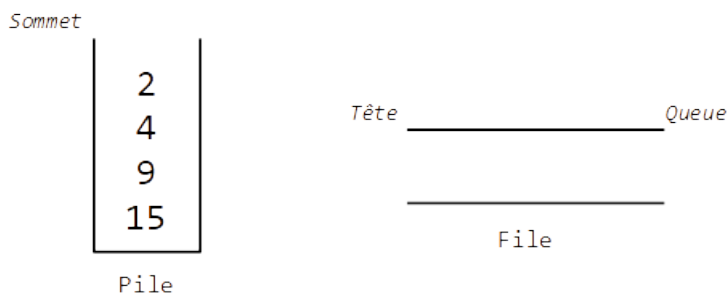
Question 1 - Recopier sur la copie le schéma ci-dessus et le compléter afin d'indiquer l'état de la pile et de la file après avoir exécuté les opérations suivantes : enfiler(12), enfiler(4), enfiler(7), empiler(defiler()), empiler(3).



Question 2a - Recopier sur la copie le schéma ci-dessus et le modifier afin d'indiquer l'état de la pile et de la file après avoir effectué deux cycles de la boucle principale (tant que la file n'est pas vide...) de cet algorithme.



Question 2b - Recopier sur la copie le schéma ci-dessus et le modifier afin d'indiquer l'état de la pile et de la file après avoir effectué l'intégralité de cet algorithme.



Question 3 - Indiquer ce que réalise cet algorithme et préciser quel en est le coût.

Cet algorithme permet de trier par ordre croissant les valeurs de la File.

Son coût est quadratique : il dépend du nombre de valeurs de la file, élevé au carré, dans le pire des cas, c'est à dire si les valeurs de la file sont déjà triées par ordre croissant de la tête vers la queue.

En effet, si n est le nombre de valeurs de la file, alors :

- la première valeur de la file sera défilée et empilée, n fois ;

- la deuxième valeur de la file sera défilée et empilée, n- fois ;
- la dernière valeur de la file sera défilée et empilée, 1 seule fois.

Le nombre d'opérations 'defiler/empiler' est donc égal à :

$$n + n-1 + n-2 + \dots + 1.$$

Il s'agit d'une suite arithmétique dont la somme se calcule de la façon suivante : $((n + 1) / 2) * n$

Par ailleurs :

- la première valeur de la file sera dépilée et enfilée, n-1 fois ;
- la deuxième valeur de la file sera dépilée et enfilée, n-2 fois ;
- la dernière valeur de la file ne sera jamais dépilée et enfilée.

Le nombre d'opérations 'depiler/enfiler' est donc égal à :

$$n-1 + n-2 + \dots + 0.$$

Il s'agit d'une suite arithmétique dont la somme se calcule de la façon suivante : $((n - 1 + 0) / 2) * n$, soit $((n - 1) / 2) * n$

Le coût de l'algorithme correspond à la somme des opérations defiler/empiler et depiler/enfiler, soit :

$$((n + 1) / 2) * n + ((n - 1) / 2) * n, \text{ soit}$$

$$n * ((n + 1) / 2) + (n - 1) / 2), \text{ soit}$$

$$n * ((n + 1 + n - 1) / 2), \text{ soit}$$

$$n * (2n / 2), \text{ soit } n^2.$$

Question 4 - Recopier sur la copie les lignes 5, 9, 13, 15 et 19 en les complétant de façon à ce que les deux fonctions puissent s'exécuter correctement.

```

3  def algo1(p, f):
4      if estVideFile(f):
5          return p, f
6
7      x = defile(f)
8      algo2(p, f, x)
9      return algo1(p, f)
10
11
12  def algo2(p, f, x):
13      if estVidePile(p) or lireSommet(p) > x:
14          empile(p, x)
15          return p, f
16
17      z = depile(p)
18      enfile(f, z)
19      return algo2(p, f, x)

```