

Piles et files

Entrées et sorties

Les listes, les tableaux ou les dictionnaires, entre autres, servent à stocker un ensemble de valeurs, en général afin de les parcourir ou de les traiter plus tard. Mais parfois, on a juste besoin de stocker les valeurs avant de pouvoir les traiter individuellement. Lorsque l'ordre de traitement dépend du moment où a été ajoutée la valeur, on distingue deux cas particuliers : premier entré premier sorti (FIFO : *first in first out*) et dernier entré premier sorti (LIFO : *last in first out*). Pour ces deux situations on utilise généralement des structures de données appelées files et piles.

Piles

Une **pile**, comme son nom l'indique, fonctionne en plaçant les valeurs au dessus les unes des autres, et seule la dernière est accessible, comme avec une pile d'assiettes.

Le premier élément qui peut être retiré est le dernier entré. C'est donc un mode de fonctionnement LIFO. L'interface classique est la suivante :

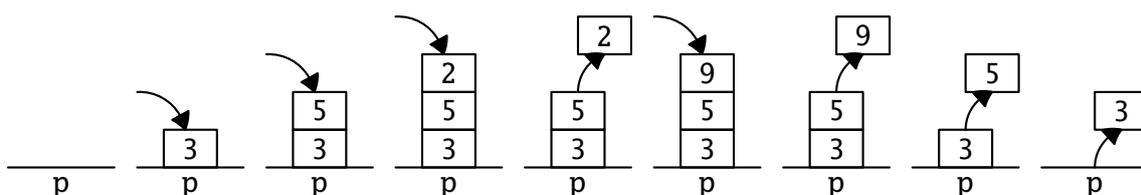
Fonction	Description
<code>cree_pile()</code>	Renvoie une nouvelle pile vide.
<code>empile(element, pile)</code>	Rajoute <code>element</code> au sommet de <code>pile</code> .
<code>depile(pile)</code>	Renvoie l'élément se trouvant au sommet de <code>pile</code> , qui ne doit pas être vide, et l'enlève.
<code>est_vide(pile)</code>	Renvoie un booléen indiquant si <code>pile</code> est vide ou non.

Voici un exemple d'utilisation :

```
>>> p = cree_pile()
>>> empile(3, p)
>>> empile(5, p)
```

```
>>> empile(2, p)
>>> v = depile(p) #v=2
>>> empile(9, p)
```

```
>>> v = depile(p) #v=9
>>> v = depile(p) #v=5
>>> v = depile(p) #v=3
```



EXERCICE 1 : Représenter l'état de la pile `p` après chacune de ces instructions, qui sont effectuées les unes après les autres.

```
>>> p = cree_pile()
>>> empile(2, p)
>>> empile(1, p)
```

```
>>> v = depile(p)
>>> empile(5, p)
>>> v = depile(p)
```

```
>>> empile(0, p)
>>> empile(6, p)
>>> v = depile(p)
```

Voici quelques situations dans lesquelles les piles peuvent être utilisées :

- **La pile d'appels des fonctions :** lorsqu'une fonction appelle une autre fonction, son exécution est interrompue, le temps que l'appel de cette nouvelle fonction se termine. L'état de la fonction initiale est alors stocké dans une pile. Dans le cas de fonctions récursives, ces appels peuvent être nombreux. L'exécution est terminée quand la pile est vidée.
- **Undo ou retour en arrière :** l'historique des modifications effectuées dans un logiciel ou l'historique des sites visités dans un onglet peuvent être stockés dans une pile permettant

ainsi de revenir en arrière pour annuler les opérations ou retourner à une page précédemment visitée. La possibilité de retourner en avant peut aussi être rajoutée avec une nouvelle pile.

Files

Les **files** fonctionnent comme les files d'attente dans les magasins. Les éléments sont retirés dans l'ordre d'arrivée. On est donc dans un mode FIFO.

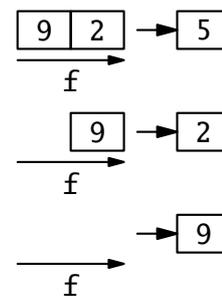
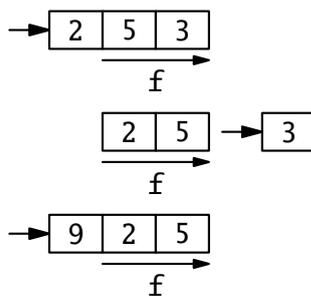
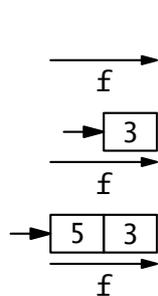
Fonction	Description
<code>creer_file()</code>	Renvoie une nouvelle file vide.
<code>enfile(element, file)</code>	Rajoute <code>element</code> à la fin de la <code>file</code> .
<code>defile(file)</code>	Renvoie l'élément se trouvant au début de <code>file</code> , qui ne doit pas être vide, et l'enlève.
<code>est_vide(file)</code>	Renvoie un booléen indiquant si <code>file</code> est vide ou non.

Voici un exemple d'utilisation :

```
>>> f = creer_file()
>>> enfile(3, f)
>>> enfile(5, f)
```

```
>>> enfile(2, f)
>>> v = defile(f) #v=3
>>> enfile(9, f)
```

```
>>> v = defile(f) #v=5
>>> v = defile(f) #v=2
>>> v = defile(f) #v=9
```



EXERCICE 2 : Représenter l'état de la file `f` après chacune de ces instructions, qui sont effectuées les unes après les autres.

```
>>> f = creer_file()
>>> enfile(2, f)
>>> enfile(1, f)
```

```
>>> v = defile(f)
>>> enfile(5, f)
>>> v = defile(f)
```

```
>>> enfile(0, f)
>>> enfile(6, f)
>>> v = defile(f)
```

Voici quelques situations dans lesquelles les files peuvent être utilisées :

- **Files d'attente :** Les files peuvent servir à chaque fois que les éléments ne peuvent pas être traités immédiatement et qu'il faut les stocker un moment. C'est par exemple le cas lorsqu'une imprimante reçoit plusieurs documents plus vite qu'elle ne peut les imprimer. Chaque document est stocké dans la file et sera imprimé dans l'ordre.
- **Pioche d'un jeu de cartes :** Dans une pioche, les cartes sont retirées les unes après les autres en partant du haut et les cartes défaussées vont au dessous.

EXERCICE 3 : Déterminer si les situations suivantes sont mieux modélisées par des piles ou des files :

- Des voitures arrêtées à un stop.
- Des crêpes sur une assiette.
- Une salle d'attente.
- Des pneus disposés en colonne.
- Des poupées russes.
- Un distributeur de gobelets, avec chargement par le haut.