



LES LISTES CHAÎNÉES

Projet informatique – P. Albers

- Le concept
- Avantages/Inconvénients
- Implémentation en python

Concept

- Collection linéaire d'éléments de données organisée comme une collection de nœuds qui contient des informations sur ce que contient ce nœud, puis un lien vers un autre nœud.
- Deux formes :
- une liste liée, qui n'a qu'une seule direction de liens entre les nœuds



- liste doublement chaînée, qui peut être liée à la fois au prochain et au dernier élément de la liste.



Avantages

- Pas de changement de numéro d'index lors des suppressions ou des insertions =>
- Temps constant sur les insertions et suppression
- Performance insertion et suppression
- Implémentation facile des Files(Queue) et (Pile Stack)

Inconvénients

- Pas d'accès aléatoire
- Utilisation de la mémoire
- Pas de traversée inverse pour liste chaînée simple

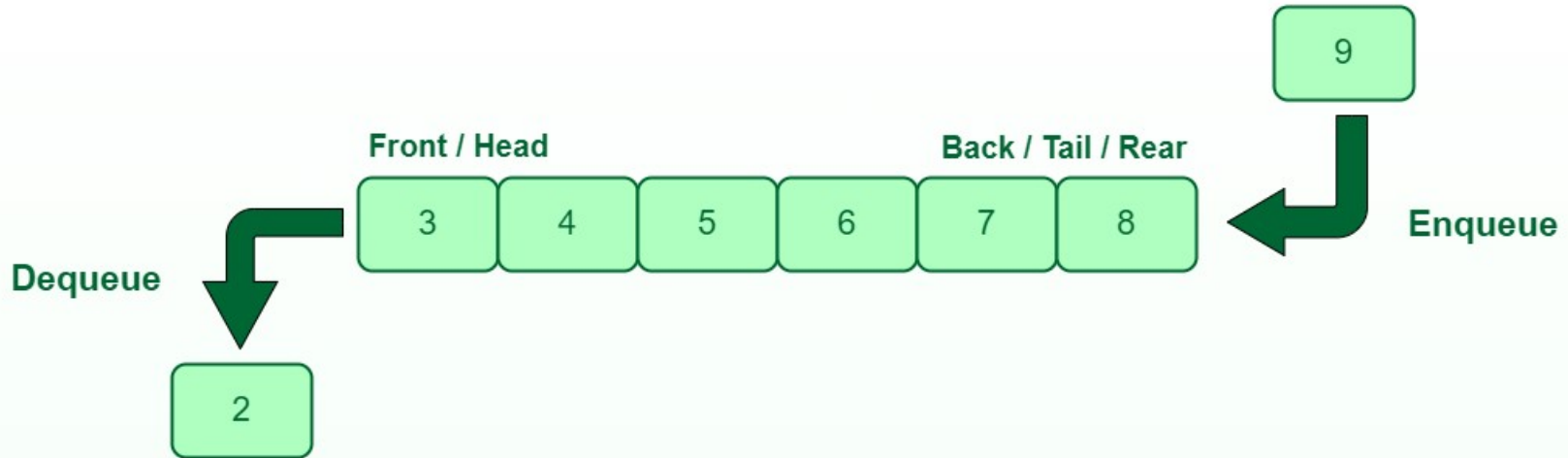
Implémentation en python

Python offre en standard depuis la version 3.6 le type `collections.deque` qui est une double liste chaînée appartenant à la famille des files (Queue)

Pour une simple liste chaînée, l'implémentation est à la charge du développeur

Qu'est ce qu'une File(Queue)

Une File(Queue) est une structure de données ouverte à la fin et au début



Utilisation deque

```
import collections  
lst = collections.deque()
```

```
# Insertion des éléments à la tête  
# Insertion des éléments à la queue
```

```
lst.append('B')  
lst.append('C')  
lst.appendleft('A')  
print(lst)  
# affiche deque(['A', 'B', 'C'])
```

```
# insertion à un indice donné  
lst.insert(2, "X")  
print(lst)  
# affiche deque(['A', 'B', 'X', 'C'])
```

```
#supprimer des éléments à la tête  
lst.pop()  
lst.popleft()  
print(lst)  
# affiche deque(['B', 'X'])
```

```
#Suppression à un index ou par le contenu  
del lst[1]  
lst.remove('B')  
print(lst)  
#affiche deque[]
```

```
#inversion  
lst = collections.deque(['A', 'B', 'X', 'C'])  
lst.reverse()  
print(lst)  
# affiche deque(['C', 'X', 'B', 'A'])
```

```
# Recherche d'un élément retourne son index  
print(lst.index('X'))  
# affiche 1
```