

QUELQUE TUTORIEL

Arrayqueue

Cette classe utilise une structure de données basée sur un tableau. La classe Queue doit garder une trace de l'endroit où début et fin de la structure est. Cela s'étendra dynamiquement sous le capot comme requis. Il s'agit d'une structure de données $O(1)$, lorsqu'il est amorti.

Il doit garder une trace de trois choses :

- pointeur vers le premier élément de la file d'attente
- nombre d'éléments en file d'attente (à utiliser à la place du dernier pointeur)
- pointeur vers la structure de données elle-même

CircularQueue

La file d'attente circulaire résout la principale limitation de la file d'attente normale. Dans une file d'attente normale, après un peu d'insertion et de suppression, il y aura un espace vide non utilisable.

La file d'attente circulaire fonctionne selon le processus d'incrémentation circulaire, c'est-à-dire que lorsque nous essayons d'incrémenter le pointeur et que nous atteignons la fin de la file d'attente, nous commençons depuis le début de la file d'attente. Ici, l'incrément circulaire est effectué par division modulo avec la taille de la file d'attente.

La file d'attente circulaire fonctionne comme suit :
deux pointeurs AVANT et ARRIÈRE

FRONT suit le premier élément de la file d'attente

REAR suit les derniers éléments de la file d'attente initialement, réglez la valeur de FRONT et REAR sur -1

1. Opération de mise en file d'attente

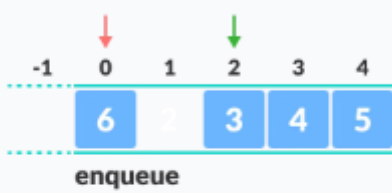
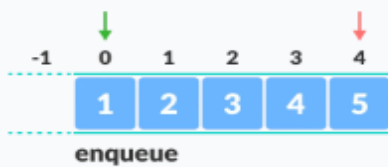
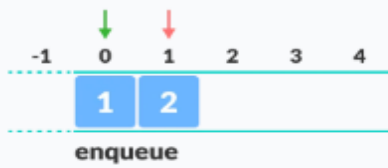
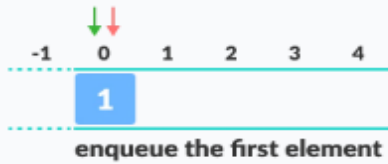
vérifier si la file d'attente est pleine pour le premier élément, réglez la valeur de FRONT sur 0 augmenter circulairement l'index REAR de 1 (c'est-à-dire que si l'arrière atteint la fin, il serait ensuite au début de la file d'attente) ajouter le nouvel élément à la position pointée par REAR

2. Opération Dequeue

vérifier si la file d'attente est vide renvoie la valeur pointée par FRONT augmenter circulairement l'index FRONT de 1 pour le dernier élément, réinitialisez les valeurs de FRONT et REAR à -1



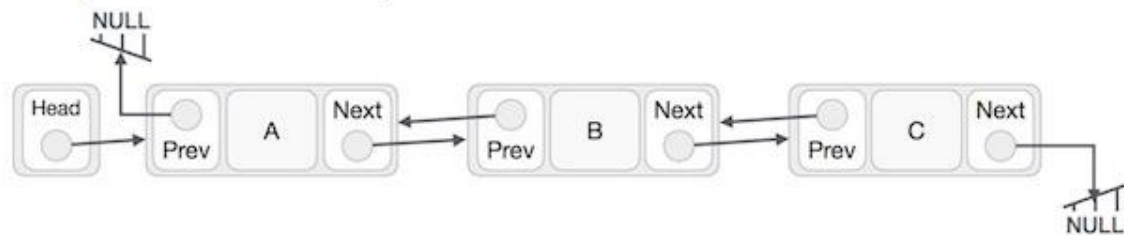
empty queue



DoublyLinked_Base

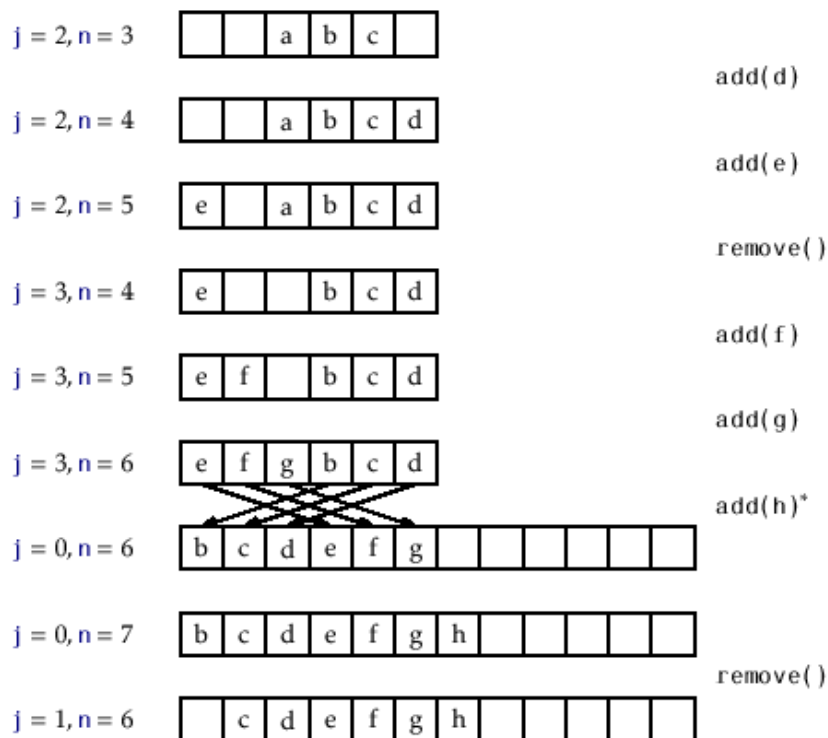
La liste doublement liée est une variante de la liste liée dans laquelle la navigation est possible dans les deux sens, en avant et en arrière facilement par rapport à la liste liée simple.

Doubly Linked List Representation



ArrayQueue

Dans cette section, nous présentons la structure de données `ArrayQueue`, qui implémente une file d'attente FIFO (premier entré, premier sorti) ;



LinkedQueue

Nous maintenons deux pointeurs, avant et arrière. L'avant pointe vers le premier élément de la file d'attente et l'arrière pointe vers le dernier élément.

enQueue() : cette opération ajoute un nouveau nœud après l'arrière et déplace l'arrière vers le nœud suivant.

deQueue () : cette opération supprime le nœud avant et le déplace vers le nœud suivant