

# Ring buffer

*and in the darkness bind them*

---

Millian POQUET

2025-04-29

# Qu'est-ce à dire que ceci ?

Différents noms pour la même chose : (ring|cyclic|circular) (buffer|queue|array)

Accès

- put : ajout d'un élément
- get : récupération d'un élément

Éléments clés

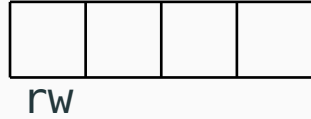
- Sémantique de file : les éléments ajoutés en premier sont récupérés en premier
- Buffer à **taille fixe** de N cases
- Utilisation de deux indices : têtes de lecture r et d'écriture w

Utilité

- Séquençage d'information entre deux composants matériels
- File de messages (synchronisée) entre deux composants logiciels

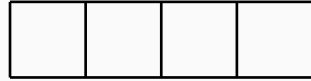
# Exemple d'exécution

`init(4,0,0)`



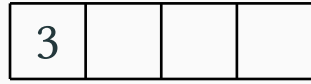
# Exemple d'exécution

`init(4,0,0)`



rw

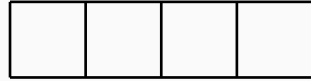
`put(3)`



r w

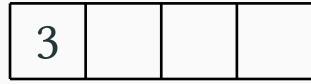
# Exemple d'exécution

init(4,0,0)



rw

put(3)



r

w

put(5)

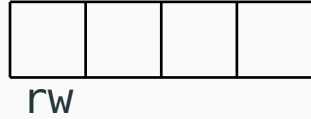


r

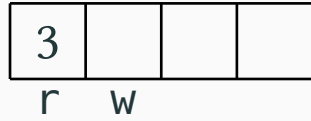
w

# Exemple d'exécution

init(4,0,0)



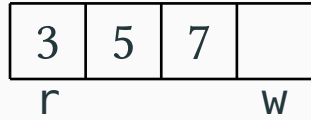
put(3)



put(5)

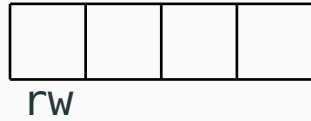


put(7)

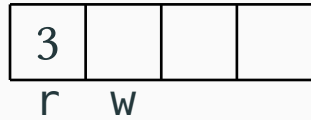


# Exemple d'exécution

init(4,0,0)



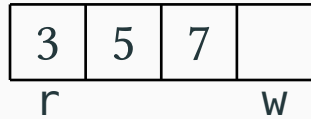
put(3)



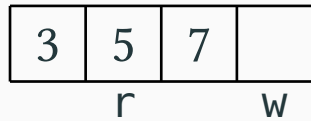
put(5)



put(7)

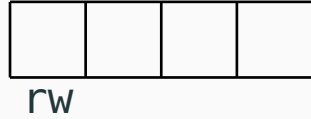


get()→3

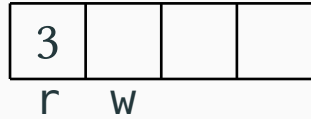


# Exemple d'exécution

init(4,0,0)



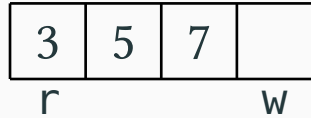
put(3)



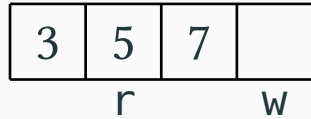
put(5)



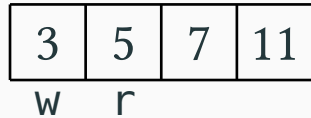
put(7)



get()→3



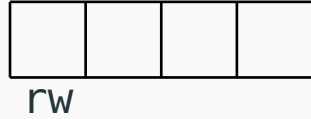
put(11)



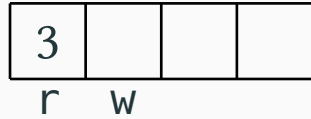


# Exemple d'exécution

init(4,0,0)



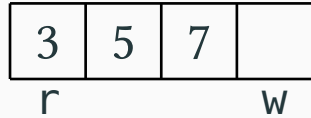
put(3)



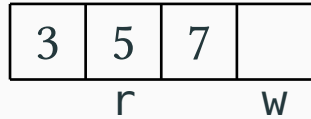
put(5)



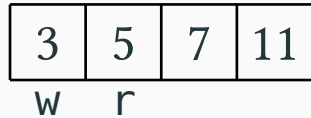
put(7)



get()→3



put(11)

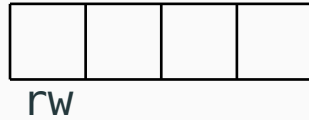


put(13)

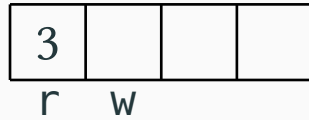


# Exemple d'exécution

init(4,0,0)



put(3)



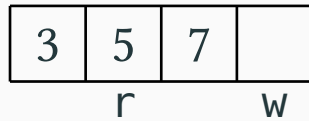
put(5)



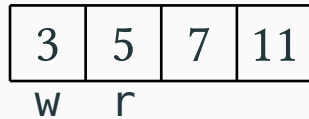
put(7)



get()→3



put(11)



put(13)

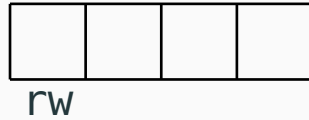


put(17)

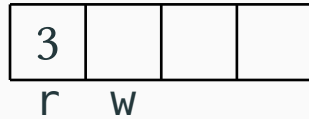
???

# Exemple d'exécution

init(4,0,0)



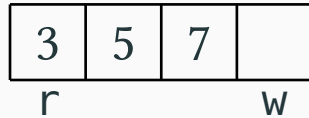
put(3)



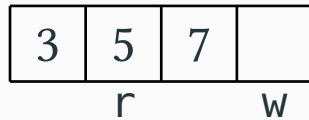
put(5)



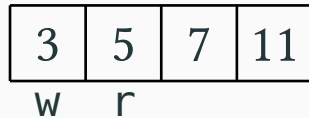
put(7)



get()→3



put(11)



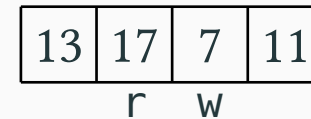
put(13)



put(17)    ???

Plusieurs choix possibles selon l'implémentation !

- put renvoie une erreur : le buffer est plein.
- put bloque jusqu'à ce que 5 soit récupéré.
- put écrase 5.



# Pseudocode d'implémentation — variante *écrase*

```
1  N = 4
2  buf = [0, 0, 0, 0]
3  w = 0
4  r = 0
5
6  def put(elem):
7      buf[w] = elem
8      w = (w + 1) % N
9
10 def get():
11     val = buf[r]
12     r = (r + 1) % N
13     return val
```