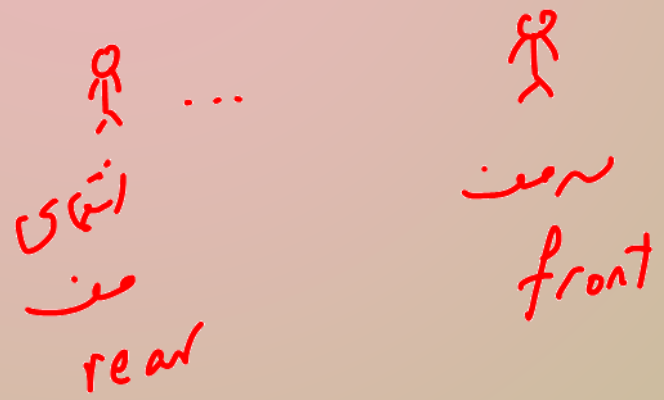
ساختمان داده ها

صف (Queue)

مدرس: غیاثی‌شیرازی
دانشگاه فردوسی مشهد

صف (Queue)

- صف نوعی لیست خطی است که عمل درج در آن به انتهای (rear) لیست و عمل حذف از آن به ابتدای (front) لیست محدود شده است.
- صف یک لیست با خاصیت FIFO (/faifo/) است.
- First In First Out



Queue ADT

empty
Test whether container is empty

size
Return size

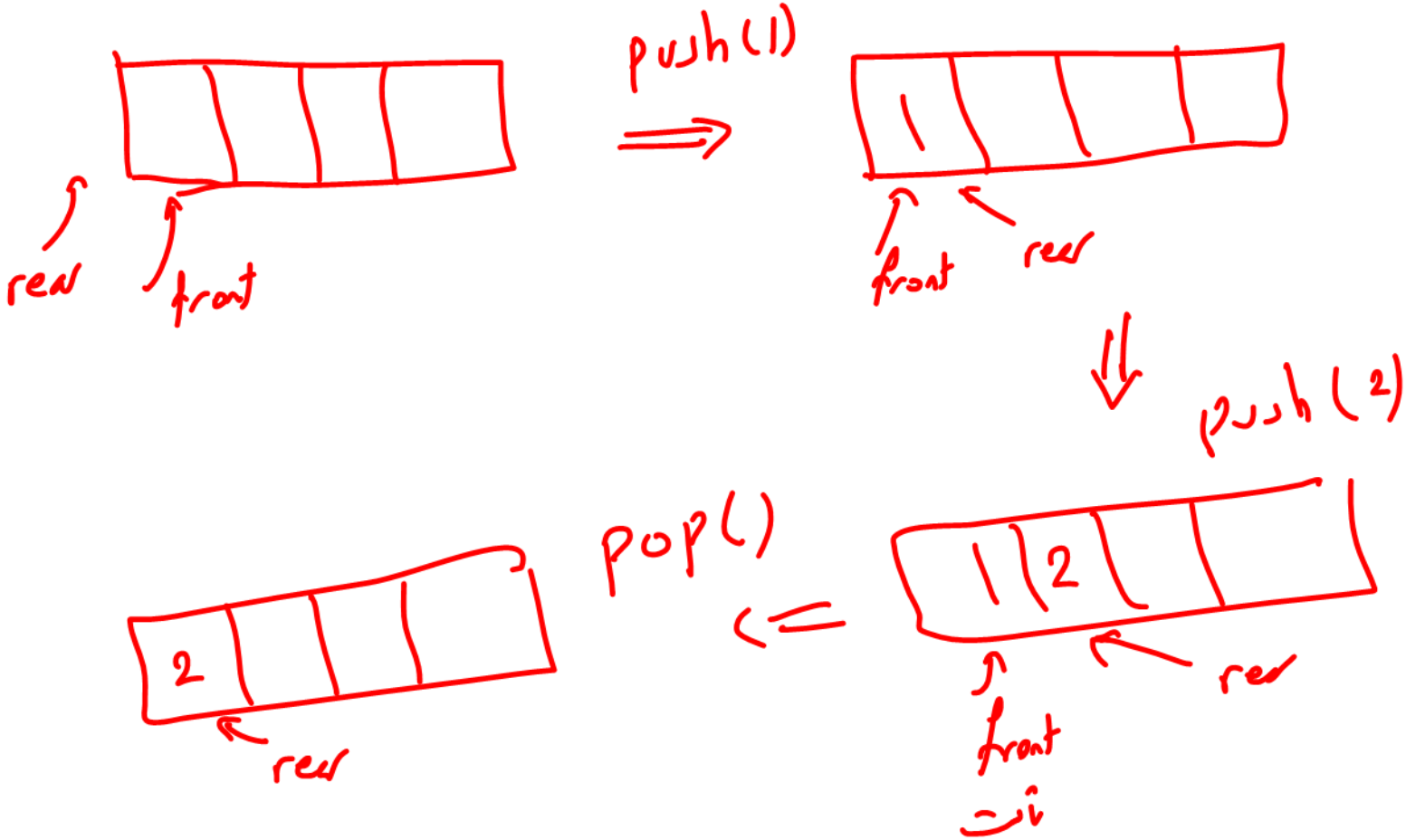
front
Access next element

back
Access last element

push
Insert element

pop
Remove next element

مزیتی که صف را با یک آرایه ساده سریع می کند



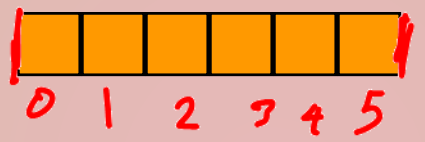
$O(1)$

مشکل: نیاز به شیفت دادن داده‌ها

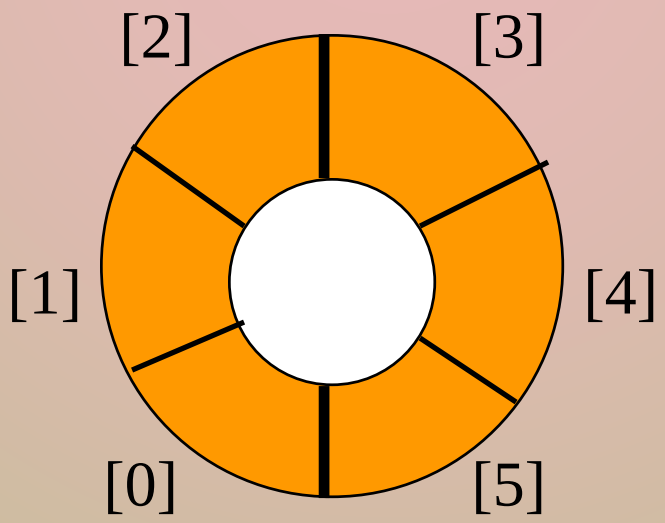
Circular Array Queue

- Use a 1D array queue.

queue[]

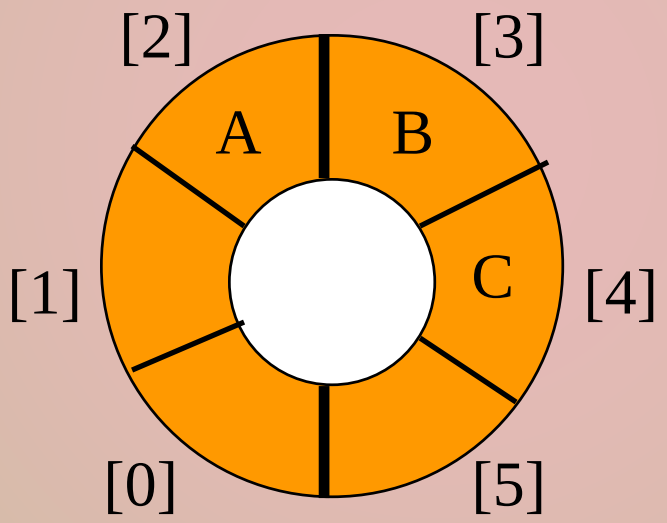


- Circular view of array.



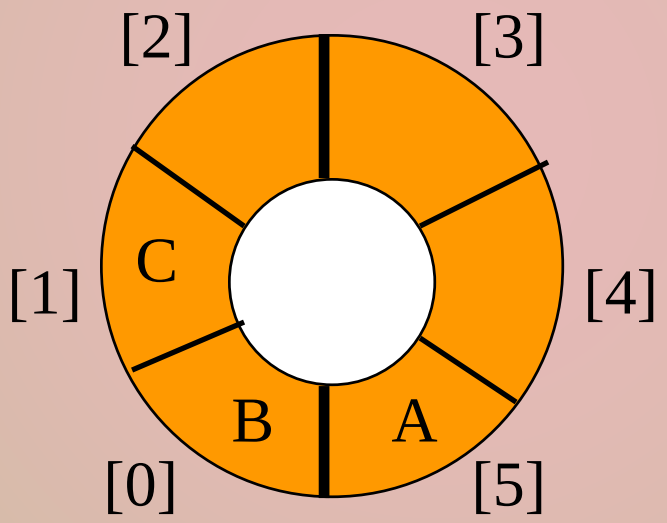
Custom Array Queue

- Possible configuration with 3 elements.



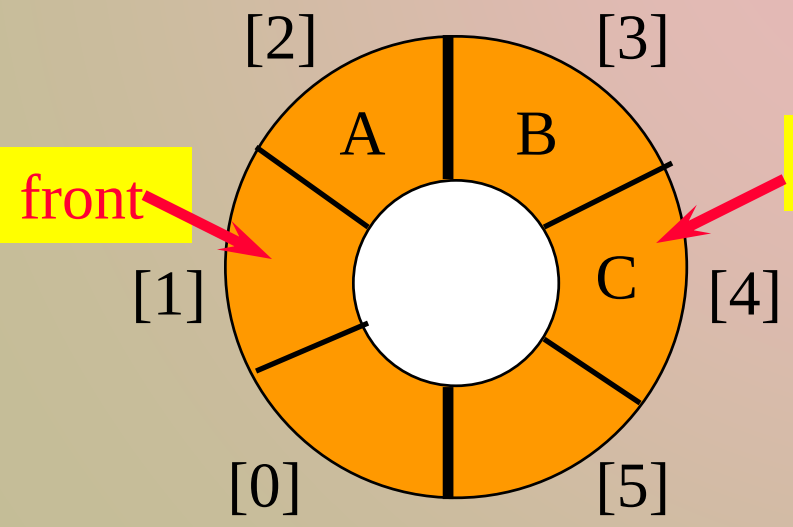
Custom Array Queue

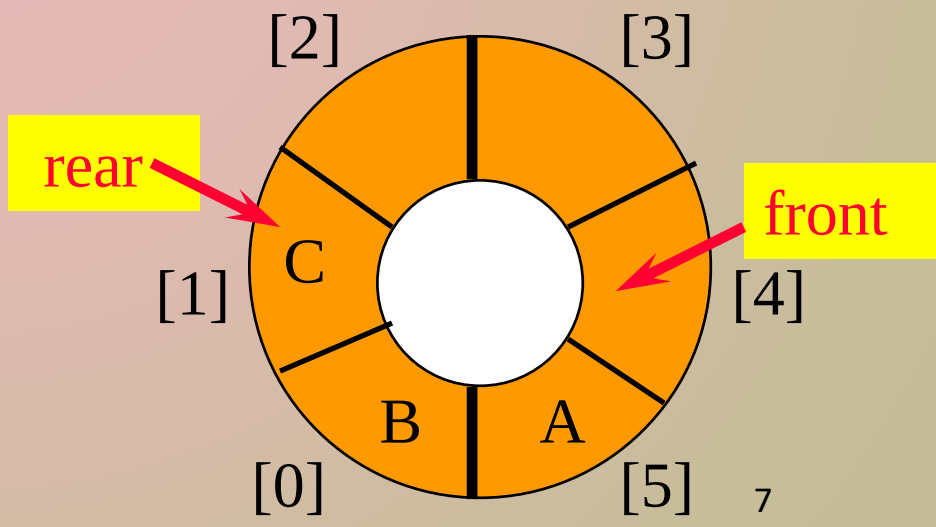
- Another possible configuration with 3 elements.



Custom Array Queue

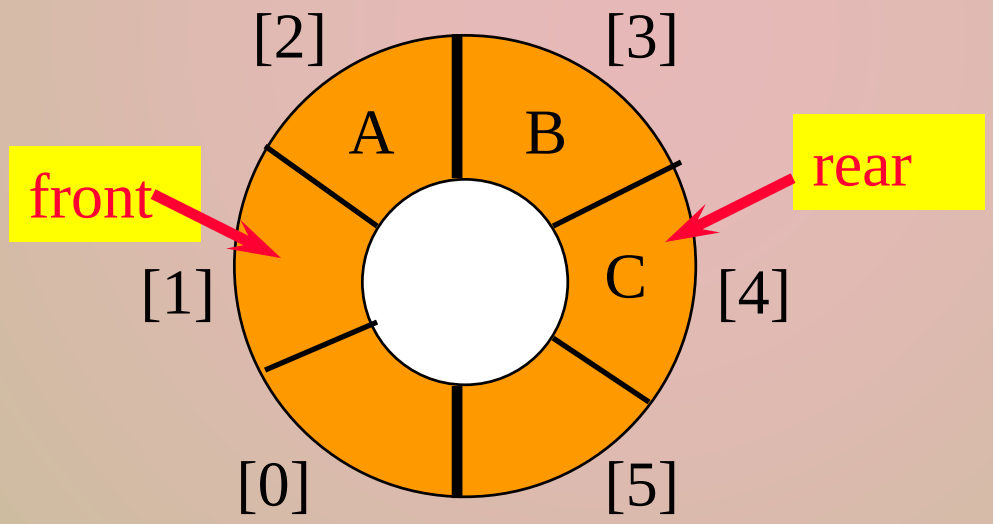
- Use integer variables front and rear.
  - front is one position counterclockwise from first element
  - rear gives position of last element





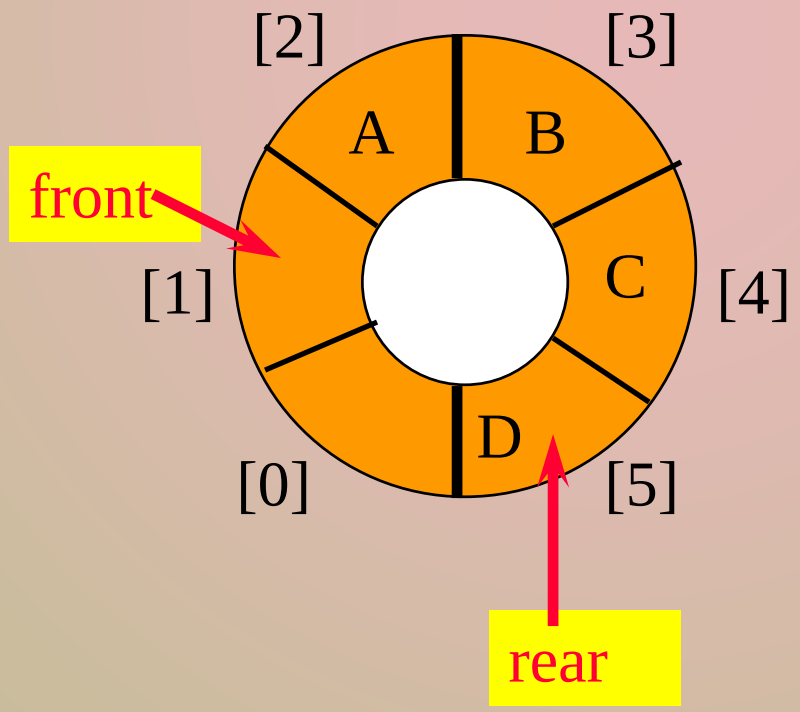
Push An Element

- Move rear one clockwise.



Push An Element

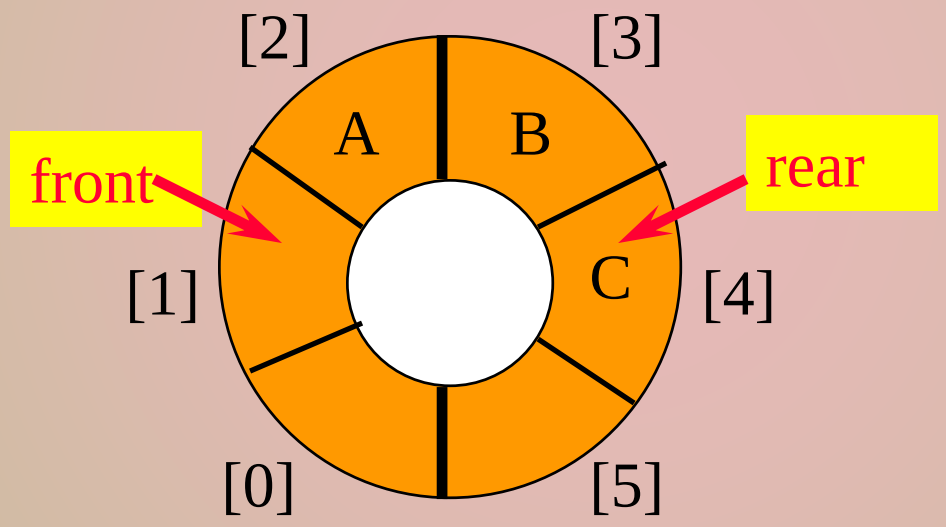
- Move rear one clockwise.
- Then put into queue[rear].



Moving rear Clockwise

- rear++;

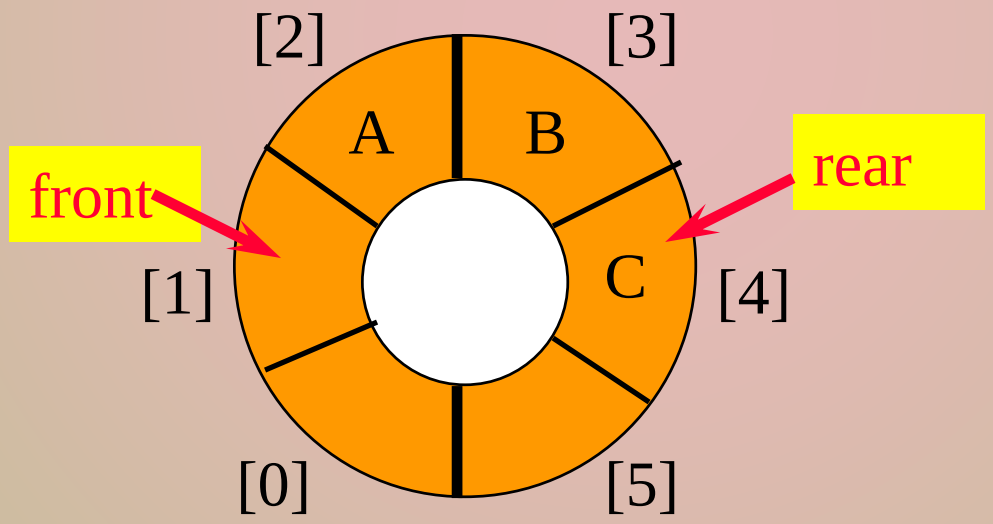
  if (rear = = capacity) rear = 0;



- rear = (rear + 1) % capacity;

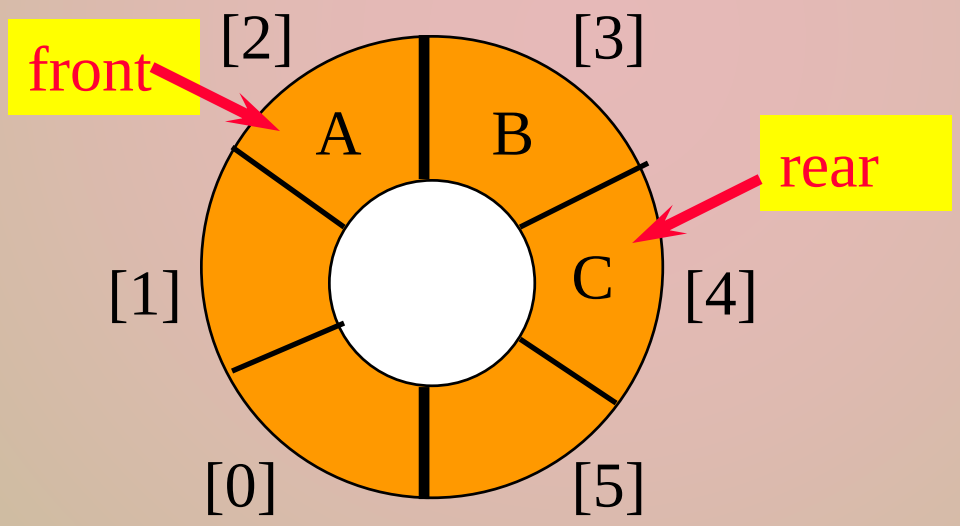
Pop An Element

- Move front one clockwise.

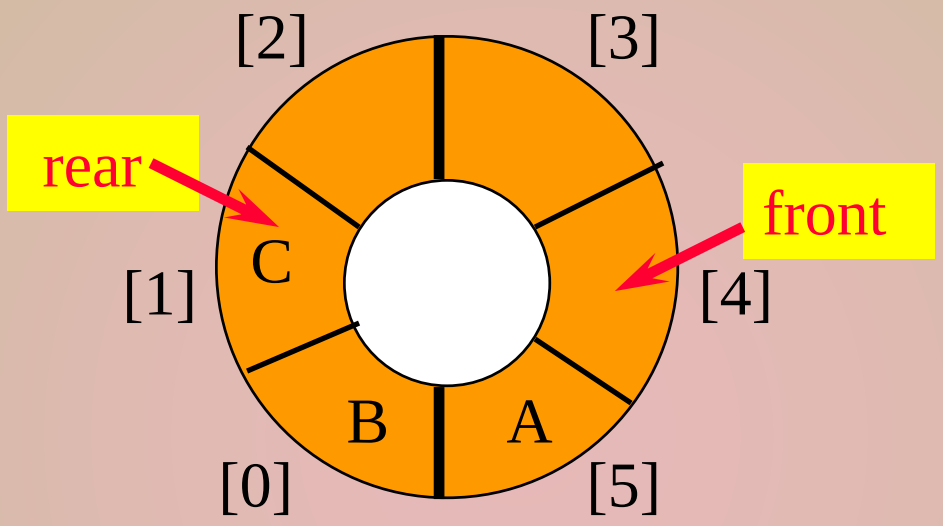


Pop An Element

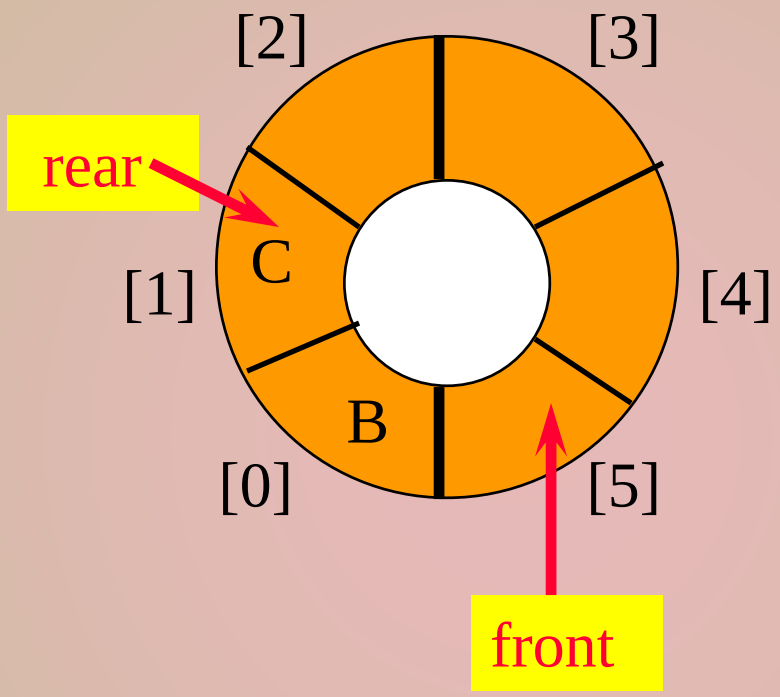
- Move front one clockwise.
- Then extract from queue[front].



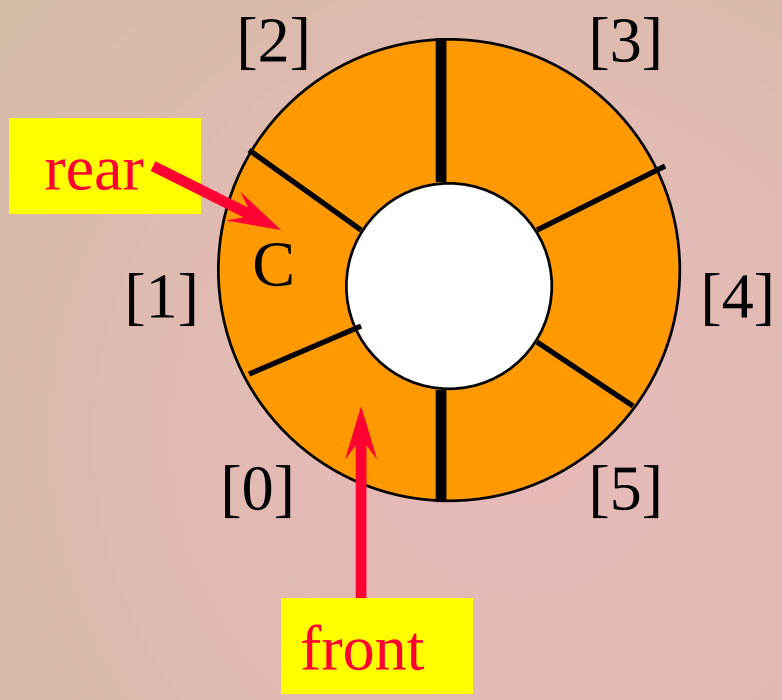
Empty That Queue



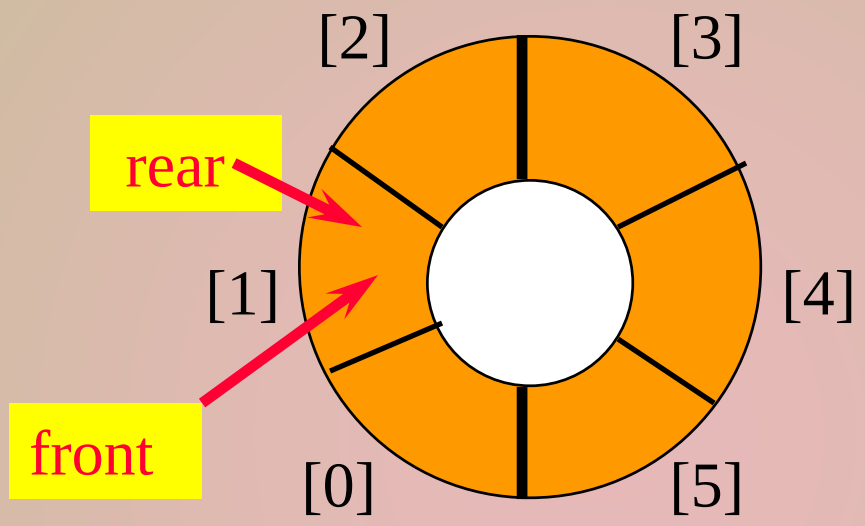
Empty That Queue



Empty That Queue

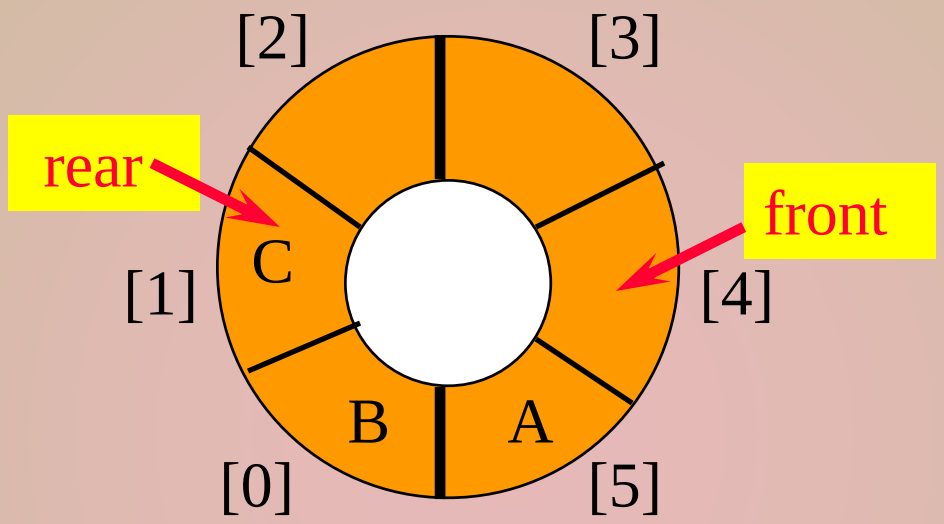


Empty That Queue

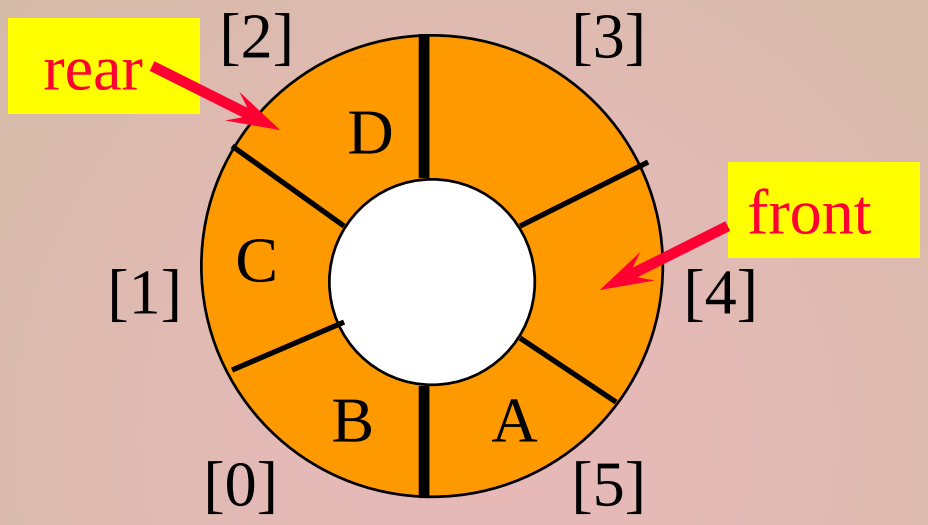


- When a series of removes causes the queue to become empty, front = rear.
- When a queue is constructed, it is empty.
- So initialize front = rear = 0.

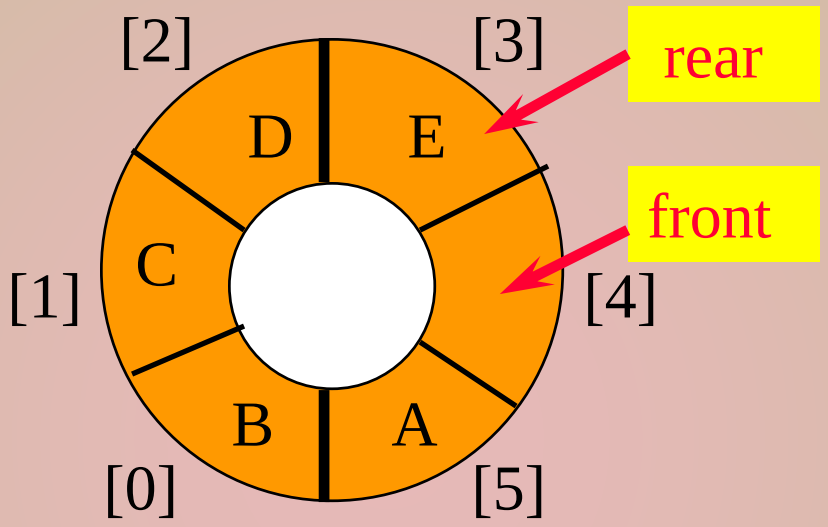
A Full Tank Please



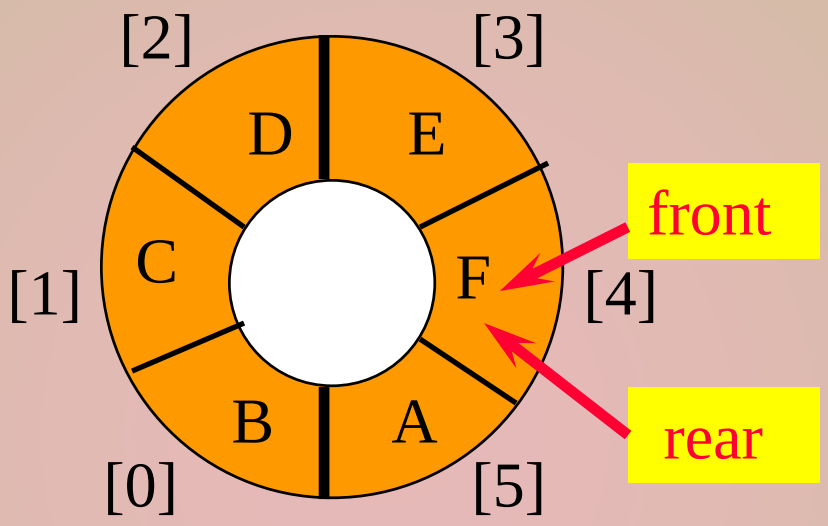
A Full Tank Please



A Full Tank Please



A Full Tank Please



- When a series of adds causes the queue to become full, front = rear.
- So we cannot distinguish between a full queue and an empty queue!

Ouch!!!!!

- Remedies.
  - Don’t let the queue get full.
    - When the addition of an element will cause the queue to be full, increase array size.
    - This is what the text does.
  - Define a boolean variable lastOperationIsPush.
    - Following each push set this variable to true.
    - Following each pop set to false.
    - Queue is empty iff (front == rear) && !lastOperationIsPush
    - Queue is full iff (front == rear) && lastOperationIsPush

Ouch!!!!!

- Remedies (continued).
  - Define an integer variable size.
    - Following each push do size++.
    - Following each pop do size--.
    - Queue is empty iff (size == 0)
    - Queue is full iff (size == arrayLength)
  - Performance is slightly better when first strategy is used.