Queue Data Structure and Types

By Dr Shantanu Pathak

Queue

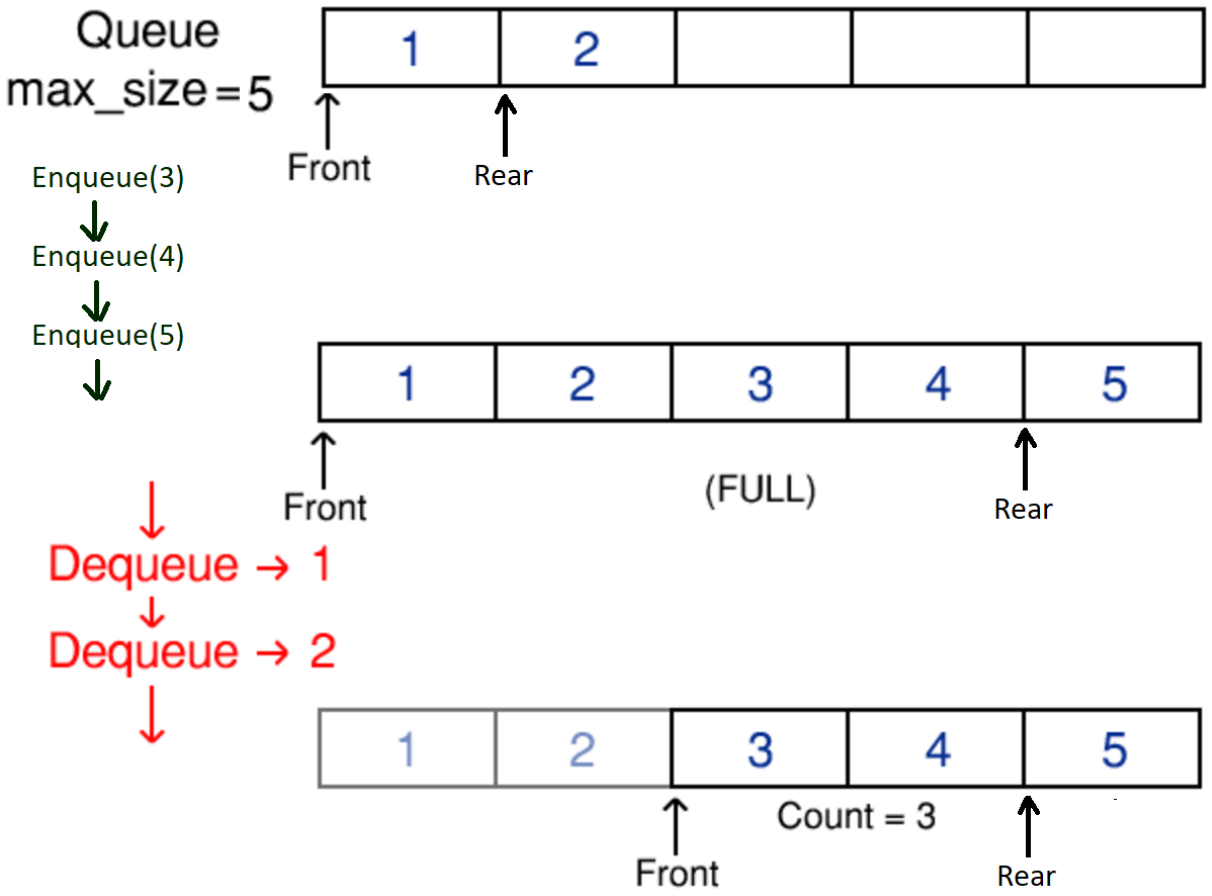
- First In First Out / Last In Last Out

- Ex.
- Queue for parking
- Queue for ticket booking

Queue Data Structure (D,F,A)

- Domains:
  - Queue of requests for ticket booking
  - Queue of process scheduling
- Functions:
  - Enqueue(): put element at rear of the queue
  - Dequeue(): remove element from front of the queue
  - IsEmpty()
  - IsFull()
- Axioms (Assumptions):
  - Element can ONLY be added at rear of the queue
  - Element can ONLY be removed from front of the queue
  - If front == rear then queue is EMPTY

Queue Working

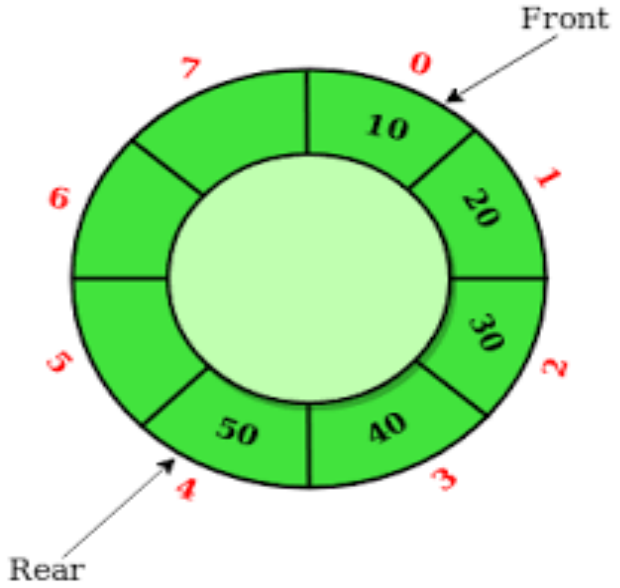


Types of Queues

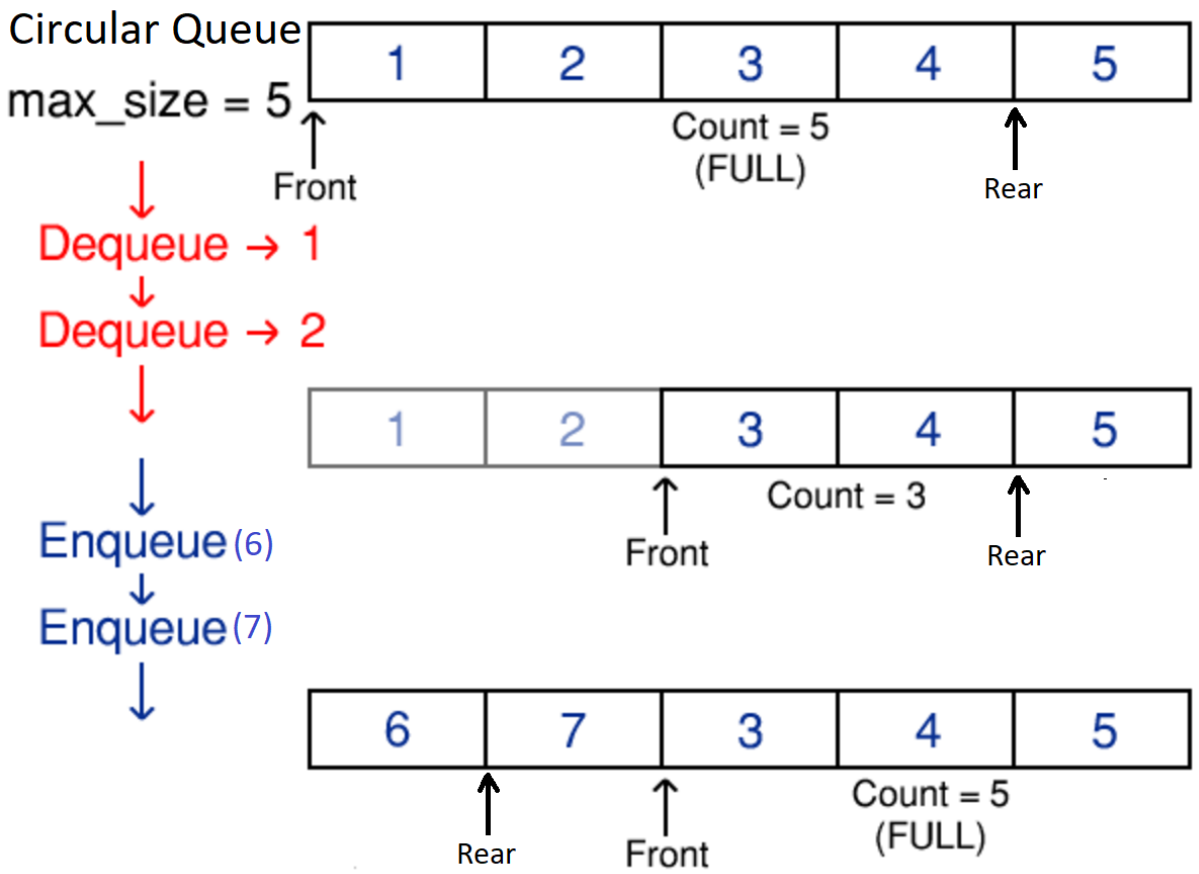
- Queue
- Double Ended Queue
- Circular Queue
- Priority Queue

Circular Queue Data Structure

- Domains:
  - Queue of process for round robin scheduling
- Functions:
  - Enqueue(): put element at rear of the queue
  - Dequeue(): remove element from start of the queue
  - IsEmpty()
  - IsFull()
- Axioms (Assumptions):
  - Element can ONLY be added at rear of the queue
  - Element can ONLY be removed from front of the queue
  - Queue EMPTY needs special handling (count)
  - Value of front and rear change in circular manner



Circular Queue Working



Circular Queue Data Structure

- Advantage
  - When queue is created using array then memory is used efficiently

- Dis-advantage
  - To detect Empty queue needs special handling

Priority Queue Data Structure

- Domains:
  - Queue of requests for loan approval
  - Queue of requests for hospital admission
  - Queue of resource allocation at server
    - (interrupt/user type/etc)
- Functions:
  - Enqueue(): insert element and priority at right place
  - Dequeue(): remove highest priority element
  - IsEmpty()
  - IsFull()
- Axioms (Assumptions):
  - All elements have priority assigned
  - Whenever element is added, it is put as per its priority