Date : 5th - 10 - 2020
Morning Session : 9am – 11.00 PM
By ~ Rohan Kumar

Topics: Queue

A queue is a useful data structure in programming. It is similar to the ticket queue outside a cinema hall, where the first person entering the queue is the first person who gets the ticket.

Queue follows the First In First Out(FIFO) rule - the item that goes in first is the item that comes out first too.

FIFO Representation of Queue

In the above image, since 1 was kept in the queue before 2, it was the first to be removed from the queue as well. It follows the FIFO rule.

In programming terms, putting an item in the queue is called an "enqueue" and removing an item from the queue is called "dequeue".

Basic Operations of Queue

A queue is an object or more specifically an abstract data structure(ADT) that allows the following operations:

- Enqueue: Add an element to the end of the queue
- Dequeue: Remove an element from the front of the queue
- IsEmpty: Check if the queue is empty
- IsFull: Check if the queue is full
- Peek: Get the value of the front of the queue without removing it

Working of Queue

Queue operations work as follows:

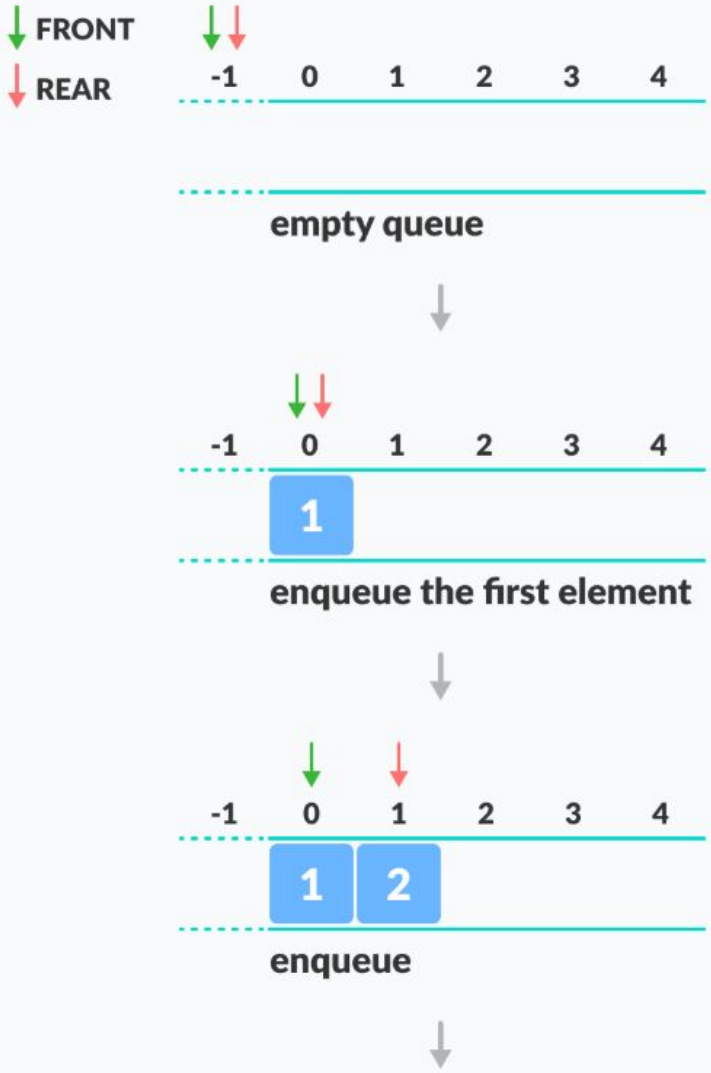
- two pointers FRONT and REAR
- FRONT track the first element of the queue
- REAR track the last elements of the queue
- initially, set value of FRONT and REAR to -1

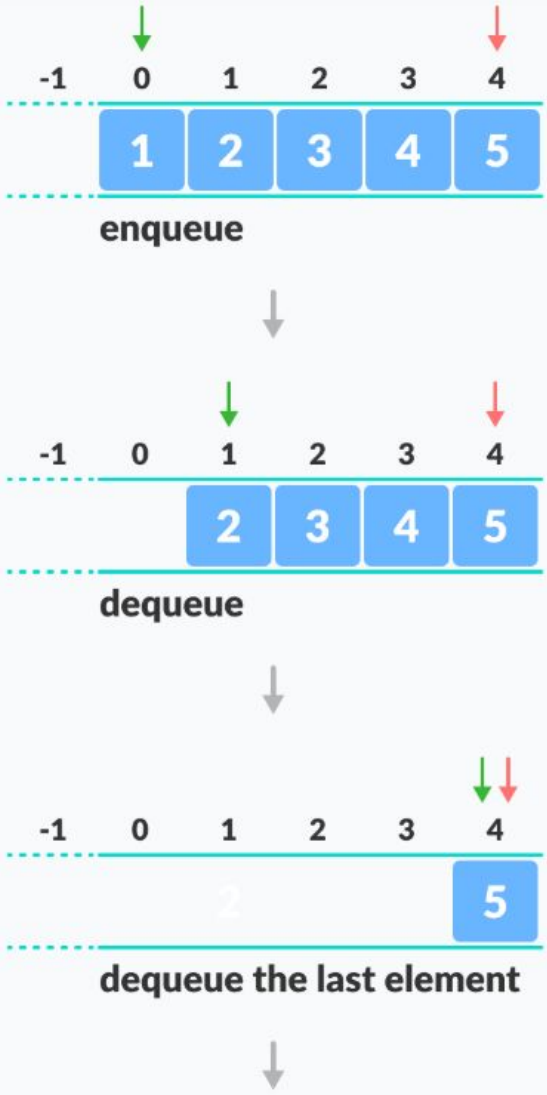
Enqueue Operation

- check if the queue is full
- for the first element, set value of FRONT to 0
- increase the REAR index by 1
- add the new element in the position pointed to by REAR

Dequeue Operation

- check if the queue is empty
- return the value pointed by FRONT
- increase the FRONT index by 1
- for the last element, reset the values of FRONT and REAR to -1


FRONT
REAR
-1
0
1
2
3
4
empty queue
1
enqueue the first element
2
enqueue



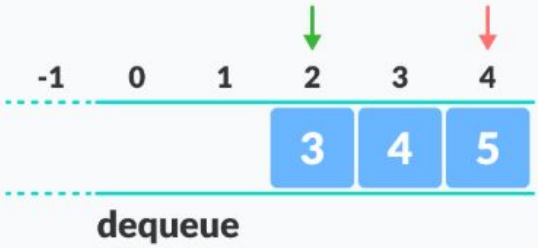
Enqueue and Dequeu Operations

Example:

```
class Queue:

    def __init__(self):
        self.queue = []

    def enqueue(self, item):
        self.queue.append(item)
    def dequeue(self):
        if len(self.queue) < 1:
            return None
        return self.queue.pop(0)
    def display(self):
        print(self.queue)

    def size(self):
        return len(self.queue)
q = Queue()
q.enqueue(1)
q.enqueue(2)
q.enqueue(3)
q.enqueue(4)
q.enqueue(5)

q.display()

q.dequeue()

print("After removing an element")
q.display()
```

Limitation of Queue

As you can see in the image below, after a bit of enqueuing and dequeuing, the size of the queue has been reduced.



Limitation of a queue

The indexes 0 and 1 can only be used after the queue is reset when all the elements have been dequeued.

After `REAR` reaches the last index, if we can store extra elements in the empty spaces (0 and 1), we can make use of the empty spaces. This is implemented by a modified queue called the circular queue.

Complexity Analysis

The complexity of enqueue and dequeue operations in a queue using an array is O(1).

Applications of Queue Data Structure

- CPU scheduling, Disk Scheduling
- When data is transferred asynchronously between two processes.The queue is used for synchronization. eg: IO Buffers, pipes, file IO, etc
- Handling of interrupts in real-time systems.
- Call Center phone systems use Queues to hold people calling them in an order

Reverse a Queue as Queue reversed using stack.

https://www.geeksforgeeks.org/reversing-a-queue/

Implement a Queue using two stacks

https://medium.com/better-programming/how-to-implement-a-queue-using-two-stacks-80772242b88c

MCQ 1:

1. A linear list of elements in which deletion can be done from one end (front) and insertion can take place only at the other end (rear) is known as a ?

a) Queue
b) Stack
c) Tree
d) Linked list

Answer: A, Queue

MCQ 2:

2. A queue is a ?

a) FIFO (First In First Out) list
b) LIFO (Last In First Out) list.
c) Ordered array
d) Linear tree

Answer: A, FIFO (First in First out) List

MCQ 3:

3. A normal queue, if implemented using an array of size MAX_SIZE, gets full when

a) Rear=MAX_SIZE-1
b) Front=(rear+1)mod MAX_SIZE
c) Front=rear+1
d) Rear=front

Answer: A, Rear = MAX_SIZE-1

MCQ 4:

6. If the elements “A”, “B”, “C” and “D” are placed in a queue and are deleted one at a time, in what order will they be removed?

a) ABCD
b) DCBA
c) DCAB
d) ABCD

Answer: A, ABCD

Resource:

https://www.programiz.com/dsa/queue