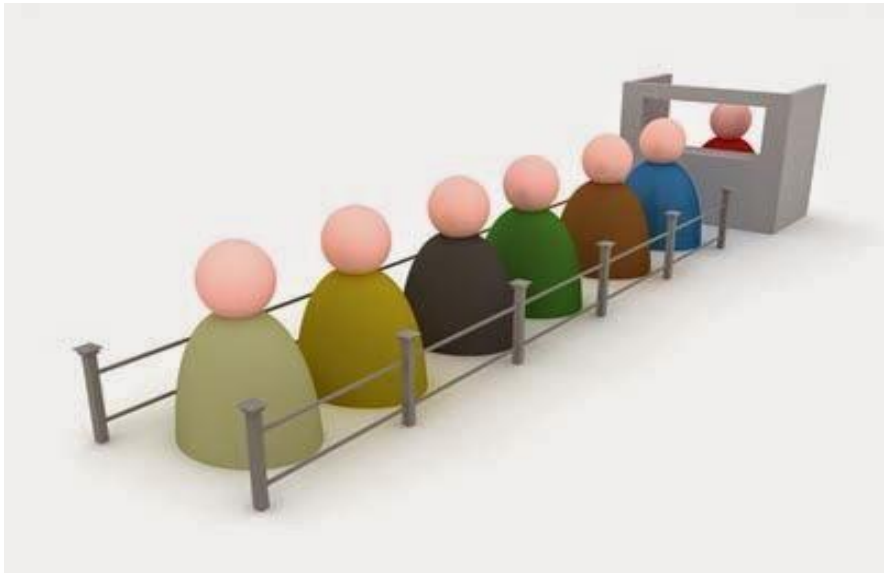
DATA STRUCTURES & ALGORITHMS

Queue

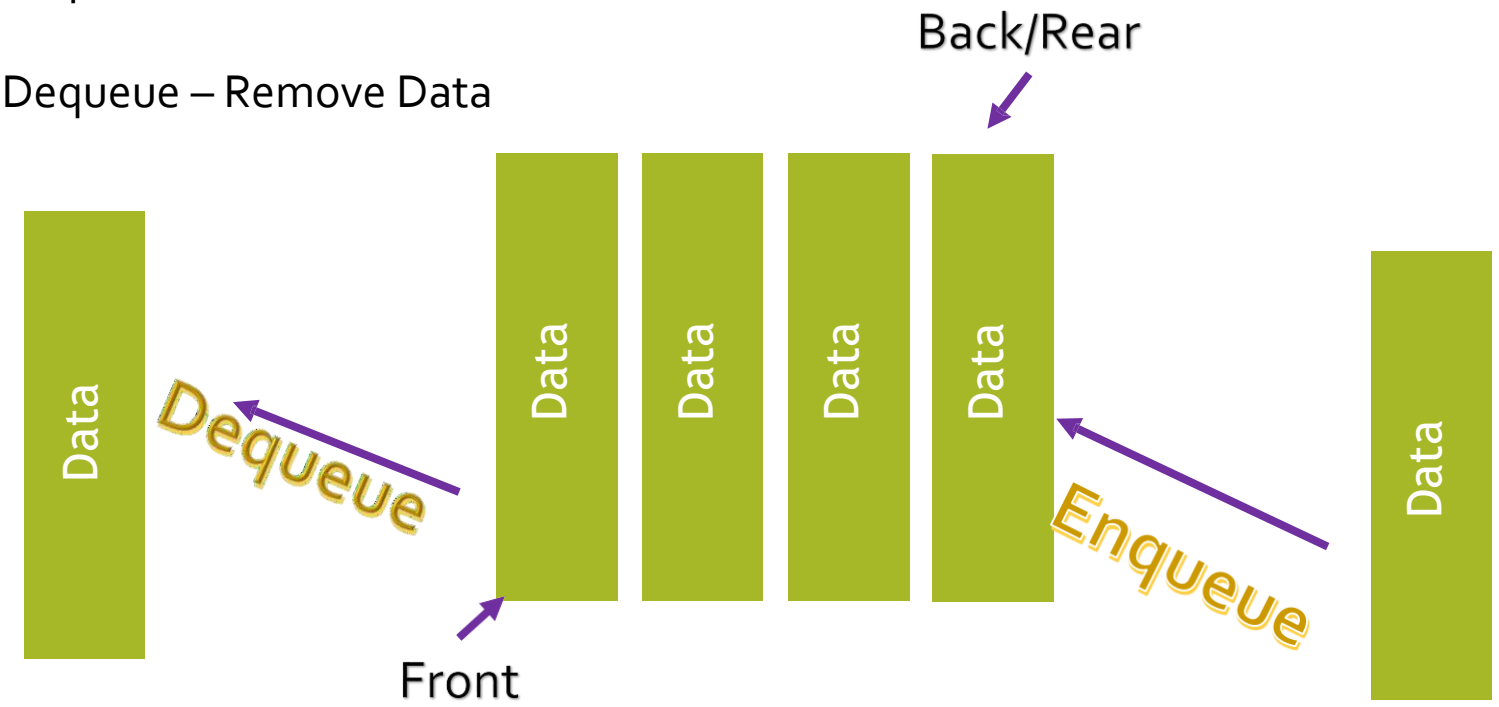
Instructor: Engr. Laraib Siddiqui

Queues



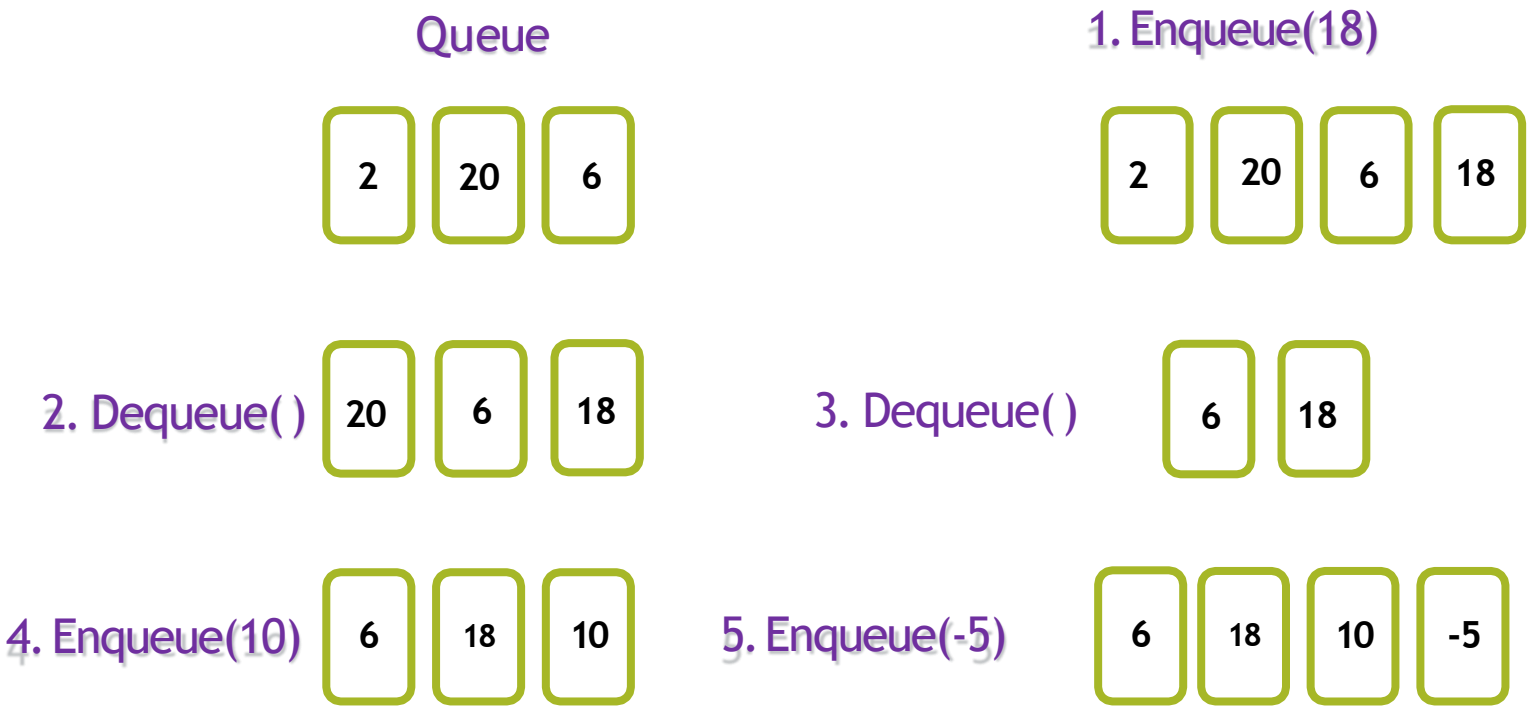
Queue

- Linear data structure.
- Follows First in First Out (FIFO Strategy).
- Primary operations
  - Enqueue – Add data
  - Dequeue – Remove Data



Example

1. Enqueue(18)
2. Dequeue( )
3. Dequeue( )
4. Enqueue(10)
5. Enqueue(-5)



Operation-Enqueue

This procedure inserts an item into a queue.

```
procedure enqueue(data)
    if queue is full
        return overflow

    end if
        rear ← rear + 1
        queue[rear] ← data
        return true

end procedure
```

Operation-Dequeue

This procedure removes an item from a queue.

```
procedure dequeue

   if queue is empty
      return underflow

   end if
      data = queue[front]
      front ← front + 1
      return true

end procedure
```

Applications

- Efficiently keep tracking of the most recently added elements.
- Web server request management where you want to first come first serve.
- Breadth first search (BFS) graph traversal.
- CPU scheduling, Disk scheduling.
- Handling of interrupts in real time systems
- Call Center phone systems use Queues to hold people calling them in order.

Types of Queue

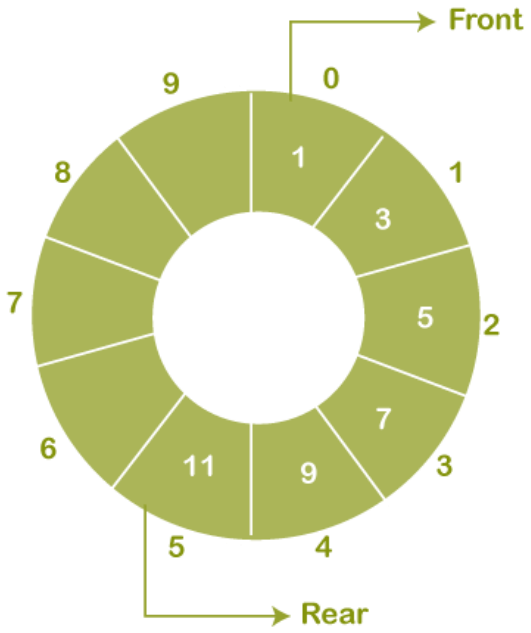
Linear Queue

insertion takes place from one end while the deletion occurs from another end.

Types of Queue

Circular queue

- Arrangement of elements in a circular fashion.
- Last node connected with first node which solves the problem of wastage of memory when removing items.
- Also known as ring buffer.



Types of Queue

Deque (Double ended Queue)

- Elements can be added or removed at either end but not in the middle.
- Also called double ended queue.

Types

1. Input restricted deque

   Insertion can be done only at one of the end, while deletion can be done from both ends.

2. Output restricted deque

   Deletion can be done only at one of the ends, while insertions can be done on both ends.

Types of Queues

Priority queue

- Collection of elements such that each element has been assigned a priority.
- Element of higher priority is processed before any element of lower priority.
- Elements with same priority are processed according to first come first serve basis.

Complexity

	Queue
Access	O(n)
Search	O(n)
Deletion	O(1)
Insertion	O(1)