Bharathidasan University

Centre for Differently Abled Persons
Khajamalai Campus
Tiruchirappalli-620 023
Tamilnadu

Bachelor of Computer Applications

(For Students with Speech and Hearing Impairment)

Course: Fundamentals of Data Structure
Unit - 2

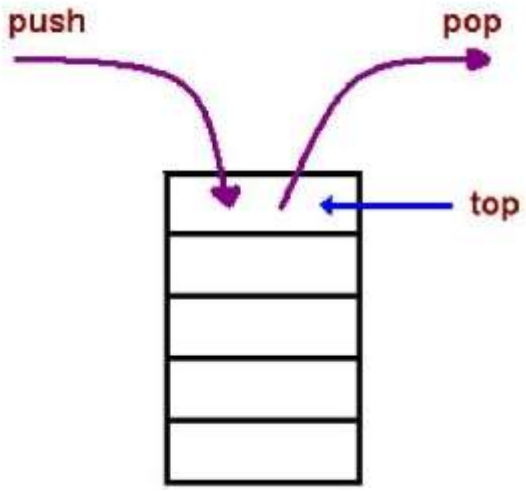
Compiled By
Dr.M.Prabavathy
(Assistant Professor)
Ms V.Vijayalakshmi

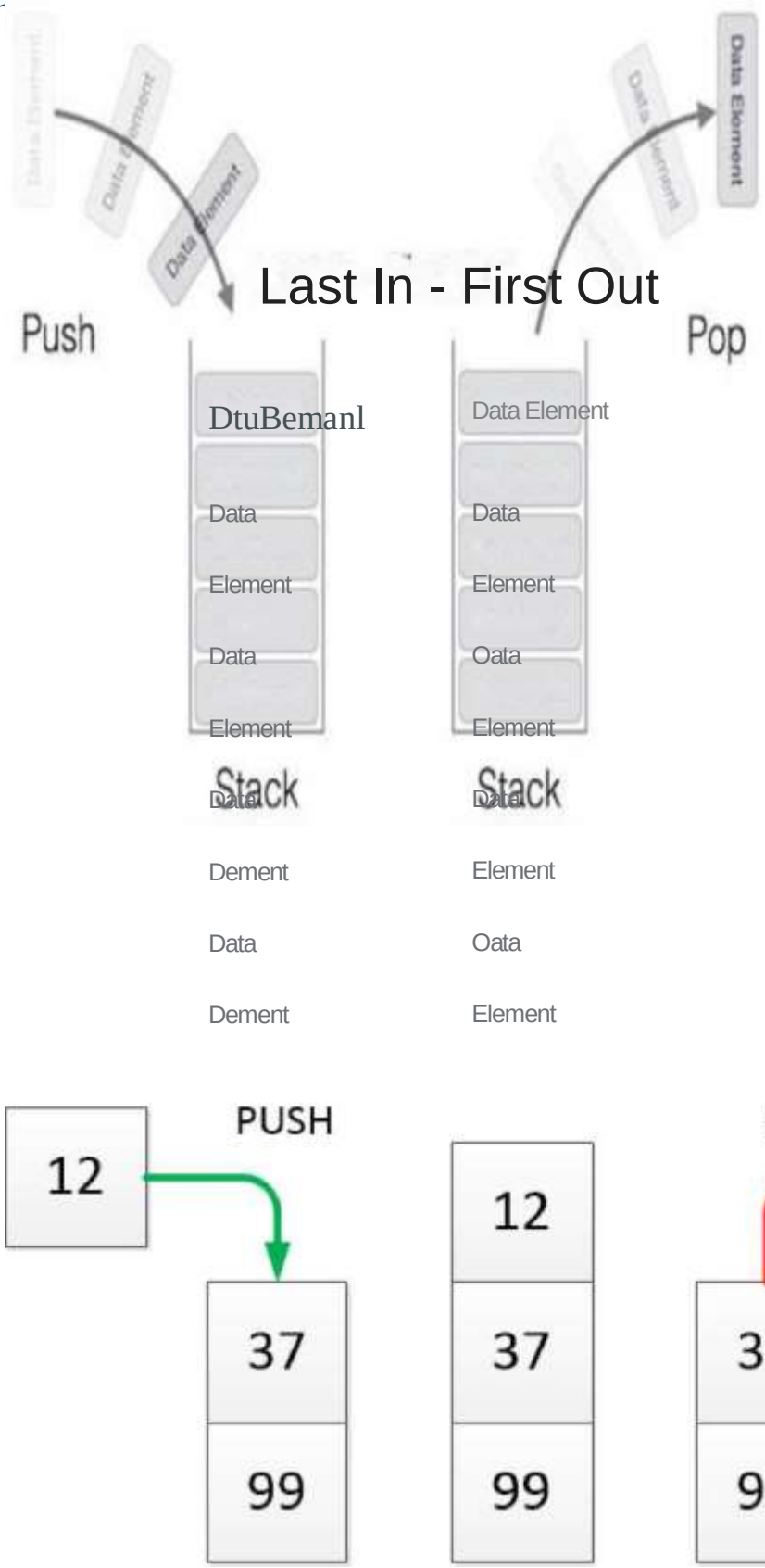
UNIT 2

Stacks:

Stack Definition

- Stack is a one type of data structure
- It is used for LIFO Order.
- **LIFO ^ LAST IN FIRST OUT**
- **PUSH** Operation is used for **add** at stack element
- **POP** Operation is used for **delete** at stack element




Last In - First Out
Push
Pop
Data Element
DtuBemanl
Data
Element
Data
Element
Stack
Data Element
Data
Element
Oata
Element
Stack
Data
Element
Oata
Element
Data
Dement
Data
Dement
12
PUSH
37
99
12
37
99
POP
12
37
99

Applications of Stack

****

1. Expression Evaluation

- Arithmetic Expression (+, *, /)
- Boolean (true or false)
- Logical Expression (&,||, !)

2. Back Tracking

- Finding the Paths, Game Playing
- Back Tracking is a form of Recursion**.**

3. Memory Management

- Memory Management is the process of controlling and co-ordinating computer memory.
- **RAM(Random Access Memory)**
- **OS(Operating Systems)**

Infix , Prefix , Postfix Conversion

****

INFIX	PREFIX	POSTFIX
a+b	+ab	ab+
a+b*c	+a*bc	abc*+
(a+b)*(c-d)	*+ab-cd	ab+cd-*
b*b-4*a*c	-*bb**4ac	bb*4a*c*-
40-3*5+1=26	+-40*3 *5* 1	40*3*5*-1+

Queues:

Queues Definition:

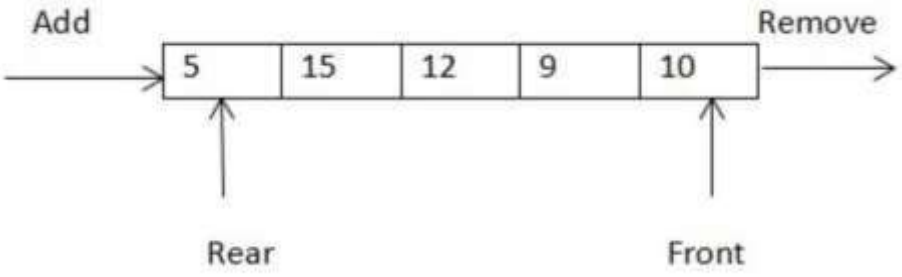
- Queue is a one type of data structure
- It is used for **FIFO** Order
- **FIFO ^ FIRST IN FIRST OUT**
- **REAR** Operation is used for **add** at stack element
- **FRONT** Operation is used for **delete** at stack element

Types of Queues:

1. **Simple Queue** :

> In a simple queue **Insertion** at the **Rear** of the list

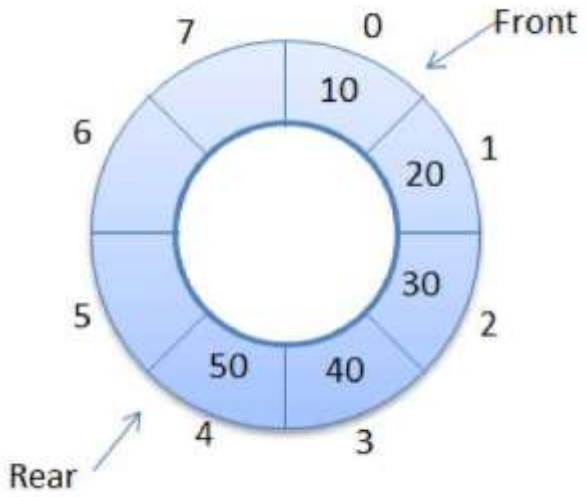
> **Deletion** occurs at the **Front** of the list



2. Circular Queue :

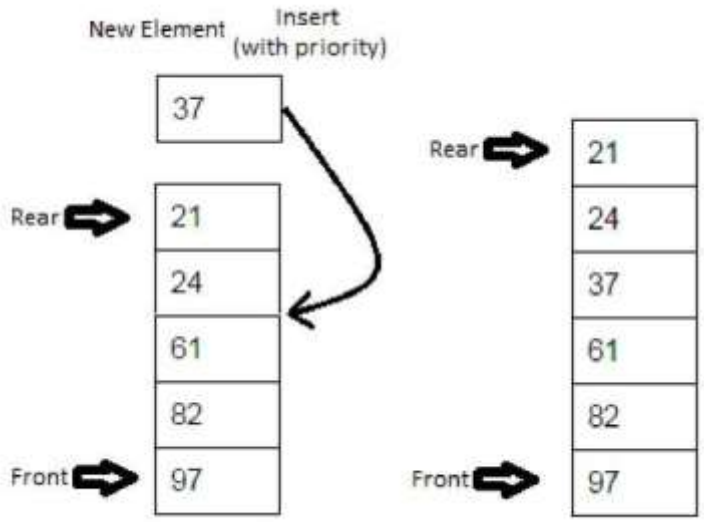
> A Circular Queue is a queue in which all nodes are treated as circular

> The First node follows the Last node



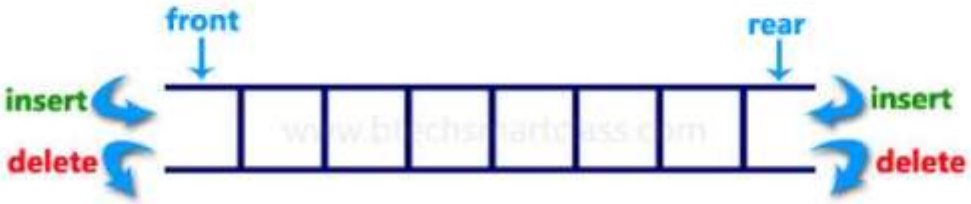
3. Priority Queue :

- The Highest Priority is removed first



4. Dequeue(Double Ended Queue) :

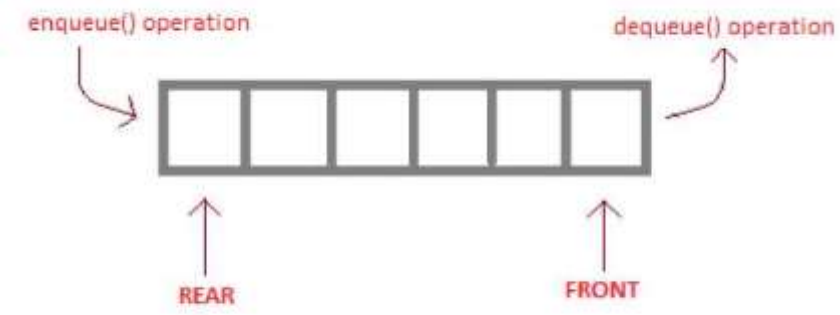
- Insertion and Deletion occur at both the end



Operations on Queues:

1. Enqueue :

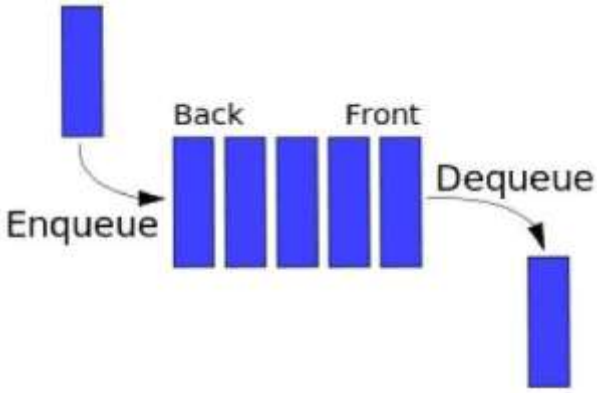
- ❖ Add Operations are possible
- ❖ Store the elements



enqueue( ) is the operation for adding an element into Queue.
dequeue( ) is the operation for removing an element from Queue .

2. Dequeue :

- ❖ Remove Operations are possible
- ❖ Access the elements

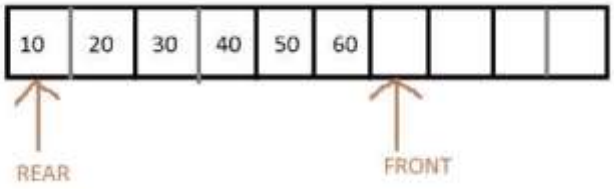


3. Initialize :

- ❖ Initial (start) stage of queue
- ❖ Procees the queue element

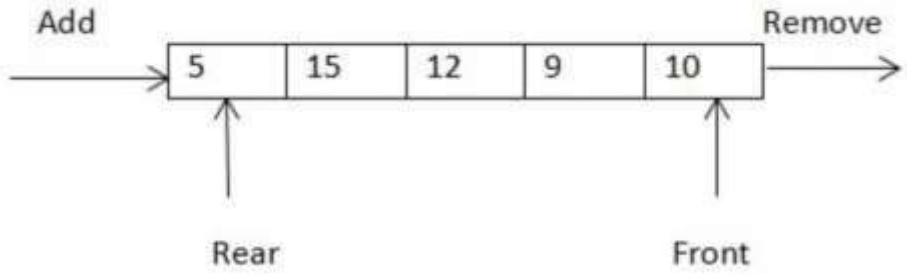
4. Is Empty :

- ❖ Queue element is not in the list
- ❖ Null of the queue



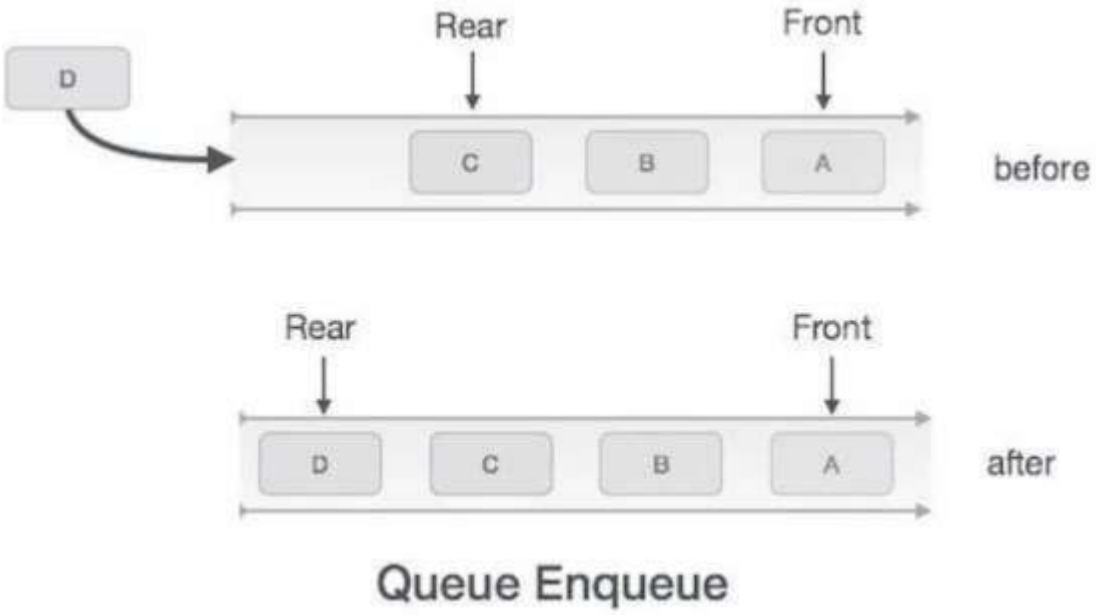
5. Is Full :

- ❖ Queue all the element are overload in the list
- ❖ Queue Space are Unavailable



6. Front :

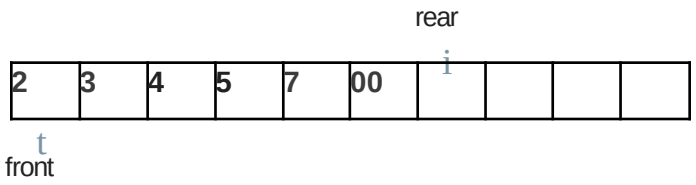
- ❖ Delete Operations are possible



Queue Enqueue

7. Peek( ) :

♦♦♦ Element at the Front of the queue without removing it.



Queues Applications:

****

1. **Destination Queues :**
   - Sending to Messages
   - Receiving a Message to Reads
2. **Administration Queues :**
   - Sending to Messages (with Acknowledgement)
   - Acknowledgement Returned
   - Connecting Applications
3. **Response Queues :**
   - Sending to Messages
   - Acknowledgement Send
   - Receiving a Message
   - Returned Acknowledgement
4. **Report Queues :**
   - Store a Message
   - Report a Message
   - Returned Message are Queuing

Thank You