



Assignment -II

Subject-DS

Div-A

Sem-I

Class –SE

Date of Issue: 22/10/2021

Date of Submission: 300/10/2021

Q.No	Question	
Q1.select correct answer		
1)	Process of inserting an element in stack is called _____ a) Create b) Push c) Evaluation d) Pop	
2)	Entries in a stack are “ordered”. What is the meaning of this statement? a) A collection of stacks is sortable b) Stack entries may be compared with the ‘<’ operator c) The entries are stored in a linked list d) There is a Sequential entry that is one by one	Mahatma Gandhi Mission's College of Engineering and Technology Kamothe, Navi Mumbai
3)	Which of the following is not the application of stack? a) A parentheses balancing program b) Tracking of local variables at run time c) Compiler Syntax Analyzer d) Data Transfer between two asynchronous process	
4)	What is the value of the postfix expression 6 3 2 4 + – * ? a) 1 b) 40 c) 74 d) -18	
5)	Here is an infix expression: 4 + 3*(6*3-12). Suppose that we are using the usual stack algorithm to convert the expression from infix to postfix notation. The maximum number of symbols that will appear on the stack AT ONE TIME during the conversion of this expression? a) 1 b) 2 c) 3 d) 4	
Q2. Choose Correct Options		
1)	In a circular queue, how do you increment the rear end of the queue? a) rear++ b) (rear+1) % CAPACITY c) (rear % CAPACITY)+1 d) rear–	

2)	With what data structure can a priority queue be implemented? a) Array b) List c) Heap d) Tree	
3)	What is a dequeue? a) A queue with insert/delete defined for both front and rear ends of the queue b) A queue implemented with a doubly linked list c) A queue implemented with both singly and doubly linked lists d) A queue with insert/delete defined for front side of the queue	
4)	Which of the following properties is associated with a queue? a) First In Last Out b) First In First Out c) Last In First Out d) Last In Last Out	
5)	What is the need for a circular queue? a) effective usage of memory b) easier computations c) to delete elements based on priority d) implement LIFO principle in queues	
6)	The postfix form of the expression $(A + B) * (C * D - E) * F / G$ is? a) $AB + CD * E - FG /**$ b) $AB + CD * E - F **G /$ c) $AB + CD * E - *F *G /$ d) $AB + CDE * - *F *G /$	
7)	The postfix form of $A * B + C / D$ is? a) $*AB / CD +$ b) $AB * CD / +$ c) $A * BC + / D$ d) $ABCD + / *$	
8)	Which data structure is used for implementing recursion? a) Queue b) Stack c) Array d) List	
9)	Circular Queue is also known as _____ a) Ring Buffer b) Square Buffer c) Rectangle Buffer d) Curve Buffer	
10)	A normal queue, if implemented using an array of size MAX_SIZE, gets full when? a) $Rear = MAX_SIZE - 1$ b) $Front = (rear + 1) \bmod MAX_SIZE$	

- c) Front = rear + 1
d) Rear = front

Q3. Answer the following questions in brief

- 1) Explain the stepwise conversion using stack for the given infix expression to the postfix expression:
 $A * (B + C) * D$

Symbol	Stack	Postfix
A		A
*	*C	A
(*C	A
B	*C	AB
+	*C+	AB
C	*C+	ABC
)	*	ABC+
*	*	ABC+*
D	*	ABC+*D
		ABC+*D*

- 2) What is recursion? Explain use of stack for recursion

Ans: Recursion is the process of defining something in terms of itself. Compilers usually execute recursive procedures by using a stack. This stack consists of all the pertinent information, including the parameter values, for each recursive call. When a procedure is called, its information is pushed onto a stack, and when the function terminates the information is popped out of the stack.

- 3) Explain priority queue. Write ADT for priority queue and state its applications

Ans: Priority Queue is a more specialized data structure than Queue. Like an ordinary queue, priority queue has the same method but with a major difference. In Priority queue items are ordered by key value so that item with the lowest value of key is at front and item with the highest value of key is at rear or vice versa. So we're assigned priority to an item based on its key value. Lower the value, higher the priority. Priority Queue is an *Abstract Data Type (ADT)* that holds a collection of elements, it is similar to a normal Queue, the difference is that the elements will be dequeued following a priority order. A real-life example of a priority queue would be a hospital queue where the patient with the most critical situation would be the first in the queue. In this case, the priority order is the situation of each patient.

[Dijkstra's Shortest Path Algorithm using priority queue](#): When the graph is stored in the form of adjacency list or matrix, priority queue can be used to extract minimum efficiently when implementing Dijkstra's algorithm.

[Prim's algorithm](#): It is used to implement Prim's Algorithm to store keys of nodes and extract minimum key nodes at every step.

[Data compression](#): It is used in [Huffman codes](#) which are used to compress data.

Artificial Intelligence : [A* Search Algorithm](#) : The A* search algorithm finds the shortest path between two vertices of a weighted graph, trying out the most promising routes first. The priority queue (also known as the fringe) is used to keep track of unexplored routes, the one for which a lower bound on the total path length is smallest is given highest priority.

Heap Sort : Heap sort is typically implemented using Heap which is an implementation of Priority Queue.

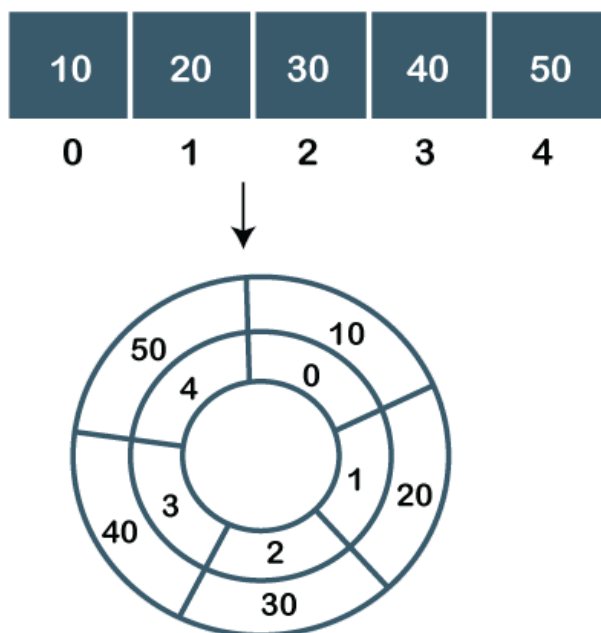
Operating systems: It is also used in Operating System for load balancing (load balancing on server), interrupt handling.

4) Explain linear queue and circular queue with suitable example. Give the advantages of circular queue over linear queue.

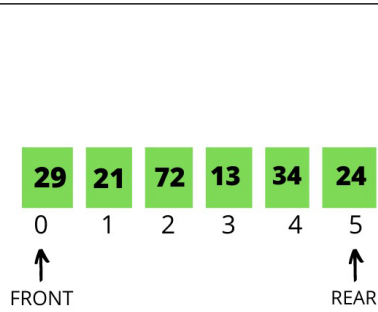
Ans: A linear queue is a linear data structure that serves the request first, which has been arrived first. It consists of data elements which are connected in a linear fashion. It has two pointers, i.e., front and rear, where the insertion takes place from the front end, and deletion occurs from the front end.



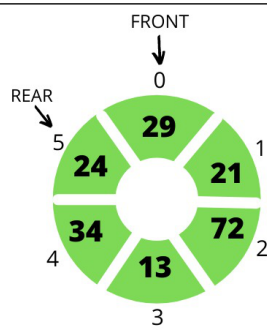
A circular queue is also a linear data structure like a normal queue that follows the FIFO principle but it does not end the queue; it connects the last position of the queue to the first position of the queue. If we want to insert new elements at the beginning of the queue, we can insert it using the circular queue data structure.



When Enqueue operation is performed on both the queues: Let the queue is of size **6** having elements **{29, 21, 72, 13, 34, 24}**. In both the queues the front points at the first element **29** and the rear points at the last element **24** as illustrated below:

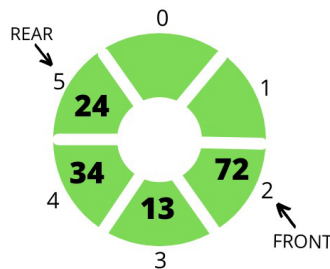
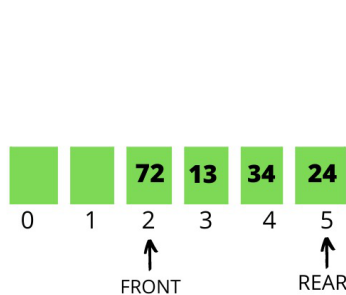


LINEAR QUEUE

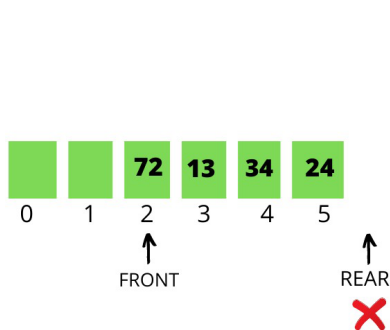


CIRCULAR QUEUE

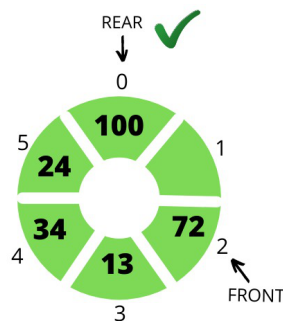
When the **Dequeue** operation is performed on both the queues: Consider the first 2 elements that are deleted from both the queues. In both the queues the front points at element **72** and the rear points at element **24** as illustrated below:



Now again enqueue operation is performed: Consider an element with a value of **100** is inserted in both the queues. The insertion of element **100** is not possible in **Linear Queue** but in the **Circular Queue**, the element with a value of **100** is possible as illustrated below:



Insertion not possible !



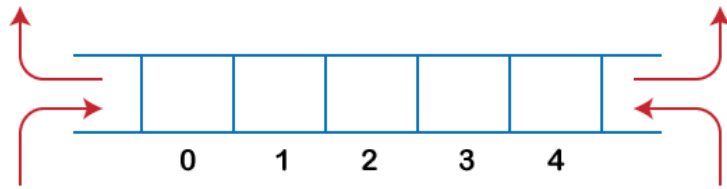
Insertion possible !

5)

Explain Dequeue with the insert and delete operations performed on it.

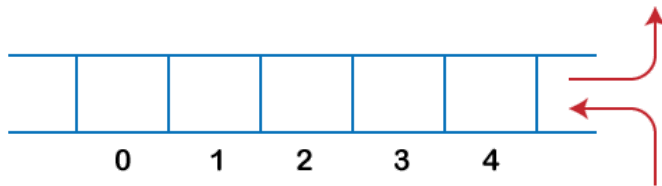
Ans: The dequeue stands for **Double Ended Queue**. In the queue, the insertion takes place

from one end while the deletion takes place from another end. The end at which the insertion occurs is known as the **rear end** whereas the end at which the deletion occurs is known as **front end**.

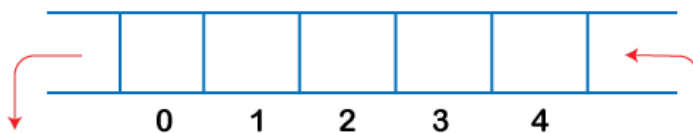


Deque is a linear data structure in which the insertion and deletion operations are performed from both ends. We can say that deque is a generalized version of the queue.

In deque, the insertion and deletion operation can be performed from one side. The stack follows the LIFO rule in which both the insertion and deletion can be performed only from one end; therefore, we conclude that deque can be considered as a stack.

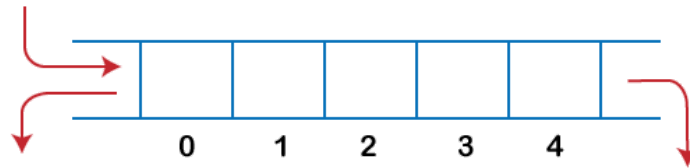


In deque, the insertion can be performed on one end, and the deletion can be done on another end. The queue follows the FIFO rule in which the element is inserted on one end and deleted from another end. Therefore, we conclude that the deque can also be considered as the queue.



There are two types of Queues, **Input-restricted queue**, and **output-restricted queue**.

1. **Input-restricted queue:** The input-restricted queue means that some restrictions are applied to the insertion. In input-restricted queue, the insertion is applied to one end while the deletion is applied from both the ends.



2. **Output-restricted queue:** The output-restricted queue means that some restrictions are applied to the deletion operation. In an output-restricted queue, the deletion can be applied only from one end, whereas the insertion is possible from both ends.

