



**Chapter -9 (23.04.2020)**

STD: XII

TIME: 30 Minutes

SUBJECT: COMPUTER SCIENCE

TOTAL MARKS: 40

Multiple Choice Questions

- Data structure stack is also known as \_\_\_\_\_ list.  
(a) First In First Out (b) First In Last Out  
(c) Last In First Out (d) All of these
- Data structure Queue is also known as \_\_\_\_\_ list.  
(a) First In First Out (b) First In Last Out  
(c) Last In First Out (d) All of these
- In a stack, all insertions take place at \_\_\_\_\_ end(s).  
(a) top (b) front (c) rear (d) any
- In a Queue, insertions take place at \_\_\_\_\_ end.  
(a) front (b) top (c) rear (d) any
- In a queue, deletions take place at \_\_\_\_\_ end.  
(a) front (b) top (c) rear (d) any
- In a stack, deletions take place at \_\_\_\_\_ end.  
(a) front (b) top (c) rear (d) any
- The terms Push and Pop are related to  
(a) Queue (b) Stack (c) Both (d) None
- Choose correct output for the following sequence of operations (• signifies top).  
push(5), push(8), pop, push(2), push(5), pop, push(1)  
(a) 8 5 2 5 1 (b) 8 5 5 2 1 (c) 2 5 5 1 (d) 5 2 1
- In which data structure element is inserted at one end called Rear and deleted at other end called Front.  
(a) Stack (b) Queue (c) Both (d) Tree
- Insertion and Deletion operations in Queue are known as \_\_\_\_\_.  
(a) Push and Pop (b) Enqueue and Dequeue  
(c) Insert and Delete (d) None
- \_\_\_\_\_ form of access is used to add and remove nodes from a queue.  
(a) LIFO, Last In First Out (b) FIFO, First In First Out  
(c) Both (a) and (b) (d) None of these
- \_\_\_\_\_ form of access is used to add/remove nodes from a stack.  
(a) LIFO (b) FIFO (c) Both (a) and (b) (d) None of these
- Stack follows the strategy of \_\_\_\_\_.  
(a) LIFO (b) FIFO (c) LRU (d) RANDOM
- Which of the following is an application of stack ?  
(a) Finding factorial (b) Reversing of a string  
(c) Infix to postfix (d) All of the above
- When a stack, implemented as an array/list of fixed size, is full and no new element can be accommodated, it is called an \_\_\_\_\_.  
(a) OVERFLOW (b) UNDERFLOW  
(c) EXTRAFLOW (d) NOFLOW
- When a stack is empty and an element's deletion is tried from the stack, it is called an \_\_\_\_\_.  
(a) OVERFLOW (b) UNDERFLOW  
(c) EXTRAFLOW (d) NOFLOW

17. If a user tries to remove an element from empty queue, it is called \_\_\_\_\_.  
 (a) Underflow (b) Empty collection  
 (c) Overflow (d) Garbage Collection
18. Pushing an element into stack already having five elements and stack has fixed size of 5, then \_\_\_\_\_ occurs.  
 (a) Overflow (b) Crash (c) Underflow (d) User flow
19. What is the term for inserting into a full queue known as ?  
 (a) overflow (b) underflow  
 (c) null pointer exception (d) program won't be compiled
20. A queue in which elements get added in empty area in the front of a queue is called \_\_\_\_\_.  
 (a) full queue (b) circular queue (c) rounded queue (d) rotated queue

### Fill in the Blanks

1. A stack is a linear structure implemented in LIFO ( \_\_\_\_\_ ) manner.
2. The technical terms for insertion-in-a-stack and deletion-from-stack are \_\_\_\_\_ and \_\_\_\_\_ respectively.
3. A queue is a linear structure implemented in FIFO ( \_\_\_\_\_ ) manner.
4. The Enqueue operation adds item at the \_\_\_\_\_ end of the queue.
5. The Dequeue operation removes the item from the \_\_\_\_\_ end of the queue.
6. Adding an element in a queue is called \_\_\_\_\_ operation.
7. \_\_\_\_\_ refers to a situation (ERROR) when one tries to enqueue/insert an item in a queue that is full.
8. \_\_\_\_\_ refers to a situation (ERROR) when one tries to dequeue/delete an item from an empty queue.
9. To use the empty space in the front of a linear queue implementation, \_\_\_\_\_ queue variation is used.
10. The \_\_\_\_\_ is a deque that allows deletions at one end only, but allows insertions at both ends.

### True/False Questions

1. Push and Pop are terms related to stack data structure.
2. Enqueue and Dequeue are terms related to queue data structure.
3. The postfix conversion of an infix expression and recursion are internally implemented through stacks.
4. The circular queue better utilises the space in a fixed size queue.
5. Pushing in a full fixed size stack results into an error.
6. Popping from an empty stack results into an error.
7. Inserting in a full fixed size queue results into an error.
8. Deleting from an empty queue results into an error.
9. For reversing the order of a list of values, queues are internally used.
10. Like circular and double-ended queues, implementation formats of stacks may also be varied.