

### # DS MCQs DAY-03:

Q. Which of the following statement is false about an array data structure?

- A. Array elements can be accessed by using random access which is faster access.
- B. Array is static
- C. Array is a linear data structure
- D. Addition and deletion operations are efficient on an array data structure.

**Answer: D**

Q. What is time complexity of addition and deletion operations on an array

- A.  $O(1)$
- B.  $O(n)$
- C.  $O(\log n)$
- D. None of the above

**Answer: B**

Q. On an array data structure searching operation can be performed efficiently in \_\_\_\_ time.

- A.  $O(1)$
- B.  $O(\log n)$
- C.  $O(n \log n)$
- D.  $O(n^2)$
- E.  $O(n)$

**Answer: B**

[ As we can apply binary search on an array => Binary Search takes  $O(\log n)$  time ].

Q. Searching operation takes \_\_\_\_ time on a linked list.

- A.  $O(1)$
- B.  $O(\log n)$
- C.  $O(n \log n)$
- D.  $O(n^2)$
- E.  $O(n)$

**Answer: E**

[ As we can apply only linear search on linked list => linear search takes  $O(n)$  time ].

Q. What is the time complexity to add node into the singly linear linked list at last position?

- A.  $O(n)$
- B.  $O(n^2)$
- C.  $O(1)$
- D.  $O(\log n)$

**Answer: A.**

- Q. Which of the following statement is false about singly linear linked list?
- A. In a SLLL, traversal can be done only in a forward direction.
  - B. In a SLLL, add and delete node at last position operations takes  $O(n)$  time.
  - C. In SLLL, add and delete node at first position operations takes  $O(1)$  time.
  - D. In SLLL, previous node of any node can be accessed from it.

**Answer: D**

- Q. Which of the following statement is false in a Linked List
- A. Linked List is a dynamic data structure.
  - B. Addition and Deletion operations are efficient and convenient in a Linked List than in an array.
  - C. Linked List elements can be accessed efficiently than array elements.
  - D. Linked List takes more space to store  $n$  elements than array.

**Answer: C**

- Q. Which of the following operations in a SCLL takes  $O(1)$  time?
- A. Add node at last position
  - B. Add node at first position
  - C. Delete node at last position
  - D. Delete node at first position
  - E. None of the above

**Answer: E**

**[ In SCLL, all operations takes  $O(n)$  time ].**

- Q. Which of the following statement is false?
- A. Linked List elements gets stored into the heap section.
  - B. Add element into a linked list at specific position takes  $O(1)$  time.
  - C. Searching operations is efficient on array than linked list.
  - D. None of the above

**Answer: B**

- Q. Which of the following statement is false about DLLL?
- A. This type of linked list can be traverse in forward as well backward direction.
  - B. Element can be added into this list at last position in  $O(1)$  time.
  - C. Element can be deleted from this list which is first position takes  $O(1)$  time.
  - D. Previous node of any node can be accessed.

**Answer: B**

- Q. Which of the following is false about DCCL?
- A. Traversal can be start from either first node or last node.
  - B. Addition and Deletion operations can be performed in  $O(1)$  time.
  - C. Searching can be done in  $O(\log n)$  time.
  - D. List can be traverse in both forward and backward direction.

**Answer: C**

Q. Which of the following data structure is used to implement depth first traversal algorithm?

- A. Array
- B. Linked List
- C. Stack
- D. Queue

**Answer: C**

Q. Which of the following is not a valid operation on stack?

- A. Push
- B. Peek
- C. Pop
- D. Top
- E. None of the above

**Answer: D**

Q. Stack data structure works in \_\_\_\_\_ manner.

- A. First In First Out
- B. First In Last Out
- C. Last In First Out
- D. Both A & C
- E. Both B & C

**Answer: E**

Q. Stack can be implemented by using

- A. Linked List
- B. Array
- C. Both A & B
- D. None of the above

**Answer: C**

Q. What is the condition to check stack is full or not in a dynamic stack?

- A.  $top == SIZE$
- B.  $top == SIZE-1$
- C.  $top == NULL$
- D. None of the above

**Answer: D**

**[ There is no stack full condition in a dynamic stack ]**

Q. Which of the following functions can be used to implement dynamic stack functionalities push( ) & pop( )?

- A. add\_last( ) & delete\_first( )
- B. delete\_first( ) & add\_last( )
- C. add\_last( ) & delete\_last( )
- D. None of the above

**Answer: C**

Q. Convert given infix expression into its equivalent postfix expression: **Infix expression is:  $(A*B)*(C/D)+E*F-G*H$**

- A.  $AB*CD/EF**+GH*-$
- B.  $AB*CD/*EF*+GH*-$
- C.  $ABCD*/*EF*+GH*-$
- D.  $AB*CD/*EF*GH+*-$

**Answer: B**