

1. _____ are those structures that have one or more pointers which point to the same type of structure, as their member.

Answers

1. Self Referential structures

2. Circulre structures

3. Data structures

4. Pointer structures

2. Consider the following operation performed on a stack of size 7.

```
Push(22);
Push(23);
Push(33);
Pop();
Pop();
Push(62);
Pop();
Push(41);
Pop();
Pop();
Push(43);
Push(44);
Pop();
Push(6);
Pop();
```

After the completion of all operation, the no. of element present on stack are _____.

3. Evaluate Postfix expression from given infix expression.

$A + B * (C + D) / (F + D) * E$

4. The prefix form of an infix expression $p+q-r*t$ is?

5. The difference between stacks and queues is in _____.

6. Which of the following is/are not the advantages of Linked List over an array?

1-The size of a linked list can be incremented at runtime which is impossible in the case of the array.

2-The List is not required to be contiguously present in the main memory, if the contiguous space is not available, the nodes can be stored anywhere in the memory connected through the links.

3-The number of elements in the linked list are limited to the available memory space while the number of elements in the array is limited to the size of an array.

7. The data structure required to evaluate a postfix expression is _____

8. What would be solution to given postfix notation: $2\ 3\ 1\ *\ +\ 9\ -$

9. How do you count the number of elements in the circular linked list?

10. What does the following function do for a given linked list with first node as head?

```
void function(struct node* head)
{
    if(head == NULL)
        return;
    function(head->next);
    printf("%d", head->data);
}
```

11. In _____ search start at the beginning of the list and check every element in the list.

12. A _____ node is a special node that is found at the beginning of the linked list.

13. In _____ type of expression operator succeeds its operands.

14. Write a suitable line instead of `/*****/`

```
void deleteList(struct node* head)
{
    struct node* tmp;
    while (head != NULL)
    {
        tmp = head;
        head = head->next;
        /*****/
    }
}
```

15. What is the time complexity of searching for an element in a circular linked list?

16. What is the functionality of the following piece of code?

```
public int function(int data)
{
    Node temp = head;
    int var = 0;
    while(temp != null)
    {
        if(temp.getData() == data)
        {
            return var;
        }
        var = var+1;
        temp = temp.getNext();
    }
    return 0;
}
```

17. The balance Factor of a node in a binary tree is ?

18. The preorder traversal sequence of a binary search tree is 29, 18, 10, 15, 23, 21, 37, 35, 45. Which one of the following is the postorder traversal sequence of the same tree?

19. Which of the following statements is not true about Singly circular linked list?

20. Which of the following is/are not correct.

Doubly circular linked list contains only one node, if _____.

i-next part of first node contains address of last node
ii-next part of first node contains address of first node
iii-prev part of last node contains null value
iv-prev part of last node contains address of first node