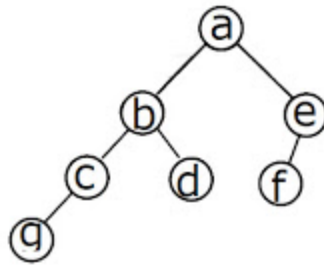1. In the balanced binary tree in the figure given below, how many nodes will become unbalanced when a node is inserted as a child of the node "g"?

(a) 1   **(b) 3**   (c) 7   (d) 8

2. Pushing an element into stack already having five elements and stack size of 5, then stack becomes
a) Overflow
b) Crash
c) Underflow
d) User flow

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. Consider the usual algorithm for determining whether a sequence of parentheses is balanced.
The maximum number of parentheses that appear on the stack AT ANY ONE TIME when the algorithm analyzes: (()(())(())) are:
a) 1
b) 2
c) 3
d) 4 or more

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

6. What is the value of the postfix expression 6 3 2 4 + – *?

a) 1

b) 40

c) 74

d) -18

7. The prefix form of an infix expression (p + q) – (r * t) is?

a) + pq – *rt

b) – +pqr * t

c) – +pq * rt

d) – + * pqrt

8. Convert the following infix expressions into its equivalent postfix expressions (A + B ∧D)/(E – F)+G

a) (A B D ∧ + E F – / G +)

b) (A B D +∧ E F – / G +)

c) (A B D ∧ + E F/- G +)

d) (A B D E F + ∧ / – G +)

9. Consider the following operation performed on a stack of size 5.

Push(1);

Pop();

Push(2);

Push(3);

Pop();

Push(4);

Pop();

Pop();

Push(5);

After the completion of all operation, the number of elements present in stack are

a) 1

b) 2

c) 3

d) 4

10. What data structure would you mostly likely see in a non recursive implementation of a recursive algorithm?

a) Linked List

b) Stack

c) Queue

d) Tree

Queues:

1. The data structure required for Breadth First Traversal on a graph is?

a) Stack

b) Array

c) Queue

d) Tree

2. A data structure in which elements can be inserted or deleted at/from both the ends but not in the middle is?

a) Queue

b) Circular queue

c) Dequeue

d) Priority queue

3. To implement a stack using queue(with only enqueue and dequeue operations), how many queues will you need?

a) 1

b) 2

c) 3

d) 4

4.   A Double-ended queue supports operations such as adding and removing items from both the sides of the queue. They support four operations like addFront(adding item to top of the queue), addRear(adding item to the bottom of the queue), removeFront(removing item from the top of the queue) and removeRear(removing item from the bottom of the queue). You are given only stacks to implement this data structure. You can implement only push and pop operations. What are the total number of stacks required for this operation?(you can reuse the stack)

a) 1

b) 2

c) 3

d) 4

5.   Consider the following code snippet. What does the sequence of statements do when executed in a sequence shown.

pt->x= (pt->x + 1) % pt->maxsize;

pt->items[pt->x] = x;

pt->size++;

a.   Linear queue -   enqueue operation

b.   **Circular queue – enqueue operation**

c.   Linear queue –   Dequeue operation

d.   Circular queue -   Dequeue operation

Note: Highlighted option is the key answer.

Thanks and Regards