

5. Each node of linked list consists of
- A. Data
B. Pointer to the next node
C. Both A and B
D. None of these
6. means how much resources the algorithm consumes.
- A. Searching
B. Sorting
C. Efficiency
D. All the above are true
7. When does the stack considered to be empty:
- A. Top = -1
B. Top = MaxStack-1
C. Top = 0
8. In Search Keys must be ordered.
- A. Binary.
B. Sequential.
C. Linear.
D. All the above are true.
9. If 'R', 'S' and 'T' are inserted into a stack in this order, the order of removal will be
- A. R, S, T.
B. T, S, R.
C. R, T, S.
D. T, R, S.
10. is a data structure consisting of a group of Nodes which together represent a sequence
- A. Linked List.
B. Stack.
C. Queue.
D. None of the above.
11. In postfix notation arithmetic expression, the operator is written the two operands?
- A. Between.
B. After.
C. Before.
12. A is an abstract model of a hierarchical structure that consists of nodes with a parent-child relation.
- A. Tree.
B. Complete binary tree.
C. Binary search tree.
D. Full binary tree.
13. Which sorting algorithm is a good algorithm to sort large number of elements
- A. Bubble Sort.
B. Selection sort.
C. Quick Sort.
D. Both A and B.
14. Search small amounts of information in efficient manner.
- A. Linear Search.
B. Binary Search.
C. Sequential Search.
D. Both A and C.
15. A is a data structure which is used to store data in a particular order.
- A. Queue
B. Linked List
C. Stack
D. None of the above

Question (3) [15 Marks]

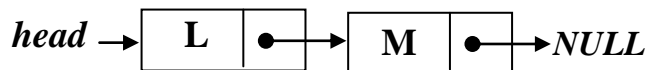
Part[1]: Define the following

(3 Marks)

- 1- Malloc () :
- 2- Sizeof () :
- 3- Free () :

Part[2]: Given the following Linked List

(7 Marks)



Write the c function that insert  **at the first of the list.**

<pre> void insertAtFirst(char item) { struct node *cur; struct node *NewNode; NewNode = (.....(node)); NewNode -> data =; NewNode -> next = NULL; if(head ==) = NewNode; } </pre>	<pre> else { cur =; while(cur -> != NULL) = cur -> next; Cur -> next =; } // End of else } // End of the function </pre>
---	---

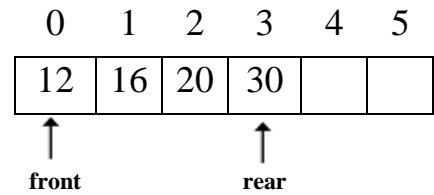
Part[3]: Complete the following code

(5 Marks)

void (char item)	else
{	{
struct node *cur;	cur =
struct node *NewNode;	while(cur -> != NULL)
NewNode = (.....(node)); = cur -> next;
NewNode -> data =;	Cur -> next =
NewNode -> next = NULL;	} // End of else
if(head ==)	} // End of the function
..... = NewNode;	

Part[3]: Having the following liner queue, perform the following actions (5 Marks)

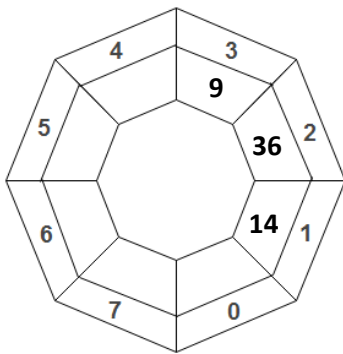
- 1- Insert (24) :
- 2- Insert (12) :
- 3- Insert (17) :
- 4- Remove () :
- 5- Remove () :
- 6- Remove () :



Finally, the Front is :, Rear is :

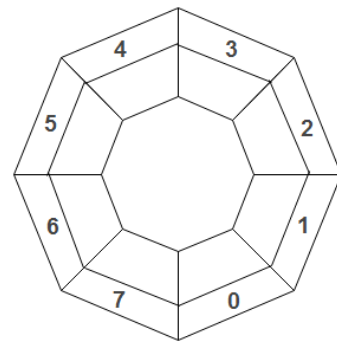
Part[4]: Having the following Circular Queue, Follow the instructions (5 Marks)

Starting point



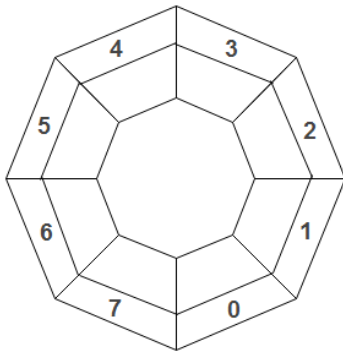
Front = 1 Rear = 3

1 Remove ()



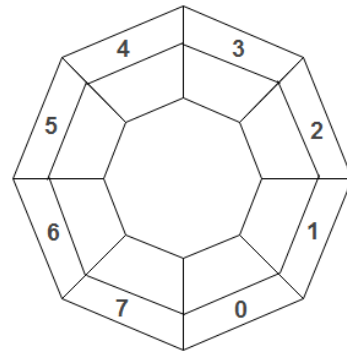
Front = Rear =

2 Remove ()



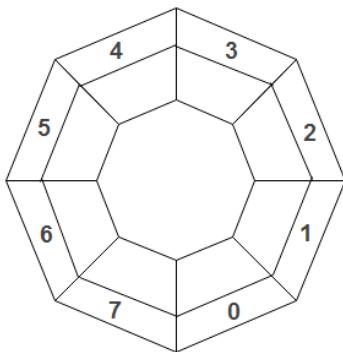
Front = Rear =

3 Remove ()



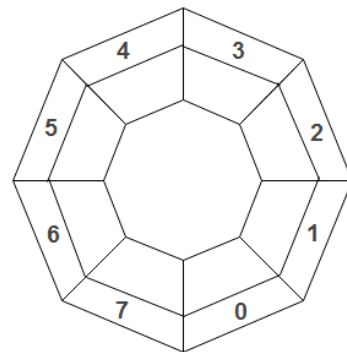
Front = Rear =

4 Insert (30)



Front = Rear =

5 Insert (15)

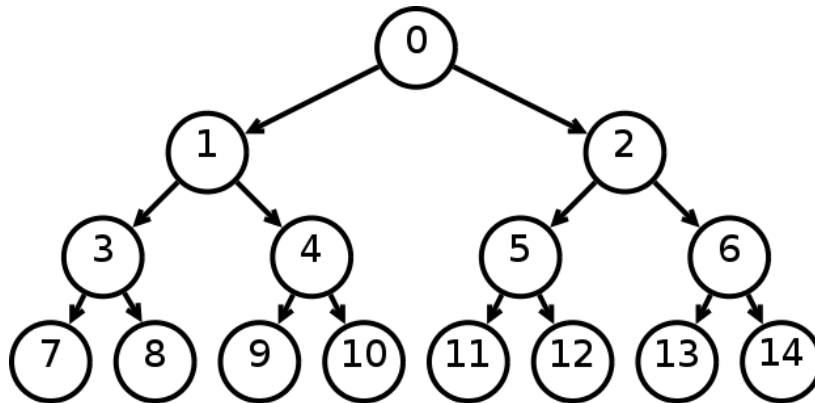


Front = Rear =

Question (5) [15 Marks]

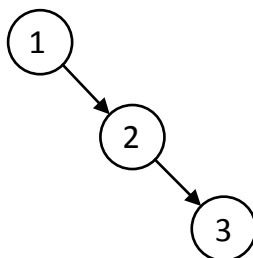
Part[1]: Having the following tree, complete the table

(9 Marks)

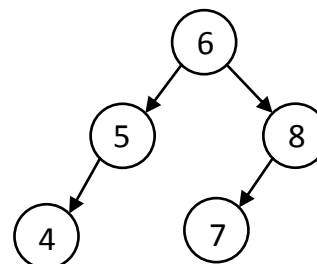


	Question	Answer
1	The root node?	
2	Parents nodes?	
3	Children nodes?	
4	Degree of node (69)?	
5	Degree of the tree?	
6	Leaf Nodes?	
7	Terminal Nodes?	
8	Depth of the tree?	
9	Type of the tree?	

Part[2]: Having the following trees, write down the traversal sequences (6 Marks)



PreOrder :
 PostOrder :
 InOrder :



PreOrder :
 PostOrder :
 InOrder :

Good Luck