(B) 4050
(C) 5040
(D) 5050

3 Stacks and Queues

1. Which one of the following is an application of Stack Data Structure?
(A) Managing function calls
(B) The stock span problem
(C) Arithmetic expression evaluation
(D) All of the above

2. Which of the following is true about linked list implementation of stack?
(A) In push operation, if new nodes are inserted at the beginning of linked list, then in pop operation, nodes must be removed from end.
(B) In push operation, if new nodes are inserted at the end, then in pop operation, nodes must be removed from the beginning.
(C) In push operation, if new nodes are inserted at the beginning, then in pop operation, nodes must be removed from the beginning.
(D) None of the above

3. Following is C like pseudo code of a function that takes a number as an argument, and uses a stack S to do processing.

```
void fun(int n){
Stack S; // Say it creates an empty stack S
while (n > 0)
{
// This line pushes the value of n%2 to stack S
push(&S, n%2);
n = n/2;
}
// Run while Stack S is not empty
while (!isEmpty(&S))
printf(”%d ”, pop(&S)); // pop an element from S and print it
}
```

What does the above function do in general?
(A) Prints binary representation of n in reverse order.
(B) Prints binary representation of n
(C) Prints the value of Logn.
(D) Prints the value of Logn in reverse order.

4. Consider the following pseudo-code using a stack

```
char stack[n];
int TOS;
TOS=-1;
input=”GATECS2019”;
char *ptr=&input;
while (ptr)
{
TOS++;
push(*ptr);
ptr++;
}
while (TOS- -!=-1)
{
print(pop());
}
```

What is the output of this program?
(A) GATECS2019
(B) GATECS201
(C) 102SCETAG
(D) 9102SCETAG

5. The following postfix expression is evaluated using a stack:
8 3 2 * - 3 * 2 + 2 4 * -
What is the final outcome of above postfix expression. Also mention the top element in stack after second * operation is completed.
(A) 1,4
(B) 0,4
(C) 1,6
(D) 0,6

6. Consider the usual way of determining whether a sequence of parentheses is balanced. The maximum number of parentheses that appear on the stack at any time when the algorithm analyzes: (()((()())(()))) is:
(A) 2
(B) 3
(C) 4
(D) 5

7. The postfix form of the expression (A + B)*(C*D-E)*F/G is?
(A) AB+CD*E–FG /**

(B) AB+CD*E–F**G/
(C) AB+CD*E–*F*G/
(D) AB+CDE*–*F*G/

8. The 8 items: A,B,C,D,E,F,G and H are pushed into a stack, one after the other starting from A.The stack is popped 7 times such that elements popped with first, third and fifth pop() operations are inserted into a queue while elements popped at second, fourth, sixth and seventh pop() operations are pushed into another temporary stack which is initially empty .Then two elements are deleted from the queue while top 3 elements are popped from the temporary stack and pushed into the queue. Finally three elements are deleted from the queue and pushed back into the original stack. Which of the following options lists all the elements of original stack from bottom to top?
(A) ABCD
(B) ADCE
(C) ACBD
(D) ADBC
(E) ABDG

9. Which of the following real world scenarios would you associate with a stack data structure?
(A) piling up of chairs one above the other
(B) people standing in a line to be serviced at a counter
(C) offer services based on the priority of the customer
(D) all of the above

10. The prefix form of A-B/(C*DˆE) is?
(A) -/*ˆACBDE
(B) -ABCD*ˆDE
(C) -A/B*CˆDE
(D) -A/BC*ˆDE

11. To evaluate an expression without any embedded function calls:
(A) One stack is enough
(B) Two stacks are needed
(C) As many stacks as the height of the expression tree are needed
(D) A Turing machine is needed in the general case

12. Which of the following statement(s) about stack data structure is/are NOT correct?
(A) Linked List are used for implementing Stacks
(B) Top of the Stack always contain the new node

(C) Stack is the FIFO data structure
(D) Null link is present in the last node at the bottom of the stack

13. Which of the following data structures can be used for parentheses matching?
(A) n-ary tree
(B) queue
(C) priority queue
(D) stack

14. Which of the following is not an inherent application of stack?
(A) Recursive function calls
(B) Reversing a string
(C) CPU Job scheduling
(D) Evaluation of postfix expression

15. Which of the following is most expensive operation in stack realized using a singly linked list?
(A) Push()
(B) Pop
(C) IsEmpty();
(D) Size();

16. If memory for the run-time stack is only 150 words, how big can N be in Factorial(N) just before encountering stack overflow? Assuming that every variable/register needs two words and a stack-frame contains at least three such fields.
(A) 75
(B) 41
(C) 26
(D) 49

17. Following sequence of operations is performed on a stack push(1),push(2),pop, push(1),push(2)pop,pop,pop,push(2),pop. The sequence of popped out elements are:
(A) 2,2,1,1,2
(B) 2,2,1,2,2
(C) 2,1,2,2,1
(D) 2,1,2,2,2

18. In evaluating the arithmetic expression 2*3-(4+5),using stacks to evaluate its equivalent postfix form, which of the following stack configuration is not possible?

(A) 4,6
(B) 5,4,6
(C) 9,6
(D) 9,3,2

19. In a stack, if a user tries to remove an element from empty stack it is called
(A) Pop
(B) Underflow
(C) Empty collection
(D) Overflow

20. Minimum number of stacks required to realize a queue completely is/are:
(A) 1
(B) 2
(C) 3
(D) 4

21. Minimum number of queues required to realize a stack completely is/are:
(A) 1
(B) 2
(C) 3
(D) 4

22. Stack is useful in:
(I) Depth First Search
(II) Breadth First Search
(III) Recursive functions calls
(IV) Postfix expression evaluation
(V) Resource allocation
(VI) Decision tree construction
(VII) Run time memory management
(VIII) Finding Paths
(IX) Backtracking algorithmic approach

Which of the following is a correct set of application of stack:
(A) I,II,III,IV
(B) I,II,IV,V,VII
(C) I,II,IV,VII,VIII,IX
(D) I,II,IV,V,VI,VII,IX

23. Consider the following statements:
I. First-in-first out types of computations are efficiently supported by

STACKS.
II. Implementing LISTS on linked lists is more efficient than implementing LISTS on an array for almost all the basic LIST operations.
III. Implementing QUEUES on a circular array is more efficient than implementing QUEUES on a linear array with two indices.
IV. Last-in-first-out type of computations are efficiently supported by QUEUES.
Which of the following is correct?
(A) (II) and (III) are true
(B) (I) and (II) are true
(C) (III) and (IV) are true
(D) (II) and (IV) are true

24. A stack contains n integer elements in it pushed in some random order. To sort these elements we are provided an additional stack. Push(), Pop() are the only operations allowed. What is the best time complexity of this task in worst case.
(A) $O(n)$
(B) $O(nlogn)$
(C) $O(n^2)$
(D) $O(n^3)$
(E) $O(2^n)$

25. Which one of the following is an application of Queue data structure?
(A) When a resource is shared among multiple users.
(B) When data is transferred asynchronously between two ends
(C) Balancing of load
(D) All of the above

26. Which of the following is not true about linked list implementation of queue?
(A) In insert() operation, if new nodes are inserted at the beginning of linked list, then in delete() operation, nodes must be removed from end.
(B) In insert() operation, if new nodes are inserted at the end, then in delete() operation, nodes must be removed from the beginning.
(C) In insert() operation, if new nodes are inserted at the beginning, then in delete() operation, nodes must be removed from the beginning.
(D) None of the above

27. A linear list of elements in which deletion can be done from one end (front) and insertion can take place only at the other end (rear) is known as a ?
(A) Queue
(B) Stack
(C) Tree
(D) Linked list

28. A Priority-Queue is implemented as a Max-Heap. Initially, it has 7 elements. The level-order traversal of the heap is given below: 15, 14, 4, 13, 10, 1, 3. Five new elements 6, 11, 14, 9, 5 are inserted in the heap in that order. And in the end one delete() operation is performed. The level-order traversal of the final heap is:
(A) 14,13,12,10,4,3,5,6,1,11,9
(B) 14,13,5,12,10,4,3,6,11,1,9
(C) 14,13,5,12,10,4,3,6,1,11,9
(D) 14,13,5,12,10,3,4,6,11,9,1

29. Consider the following pseudo-code for a queue of integers.

```
void fun(int n)
{
Queue q = new Queue();
q.enqueue(0);
q.enqueue(1);
for (int i = 0; i < n; i++)
{
int a = q.dequeue();
int b = q.dequeue();
q.enqueue(b);
q.enqueue(a + b);
ptint(a);
}
}
```

(A) Prints numbers from 0 to n-1
(B) Prints numbers from n-1 to 0
(C) Prints first n Fibonacci numbers
(D) Prints first n Fibonacci numbers in reverse order

30. In BFS of a Graph, which of the following data structure is used?
(A) Stack
(B) Queue
(C) Linked list
(D) Array

31. Suppose implementation supports an instruction REVERSE, which reverses the order of elements on the stack, in addition to the PUSH and POP instructions. Which one of the following statements is TRUE with respect to this modified stack?
(A) A queue cannot be implemented using this stack.
(B) A queue can be implemented where ENQUEUE takes a single instruction and DEQUEUE takes a sequence of two instructions.

(C) A queue can be implemented where ENQUEUE takes a sequence of three instructions and DEQUEUE takes a single instruction.
(D) A queue can be implemented where both ENQUEUE and DEQUEUE take a single instruction each.

32. Suppose you are given an implementation of a queue of integers. The operations that can be performed on the queue are:
I. isEmpty (Q) — returns true if the queue is empty, false otherwise.
II. delete (Q) — deletes the element at the front of the queue and returns its value.
III. insert (Q, i) — inserts the integer i at the rear of the queue.
Consider the following function:

```
void f (queue Q) {
int i ;
if (!isEmpty(Q)) {
i = delete(Q);
f(Q);
insert(Q, i);
}
}
```

What operation is performed by the above function f ?
(A) Leaves the queue Q unchanged
(B)Reverses the order of the elements in the queue Q
(C) Deletes the element at the front of the queue Q and inserts it at the rear keeping the other elements in the same order
(D) Empties the queue Q

33. A standard queue, when implemented using an array of size as *maxsize*, is said to be full when:
(A) Rear = *maxsize* - 1
(B) Front = (rear + 1)mod *maxsize*
(C) Front = rear + 1
(D) Rear = front

34. Which of the following option is not correct?
(A) If the queue is implemented with a linked list, keeping track of a front pointer, Only rear pointer s will change during an insertion into an non-empty queue.
(B) Queue data structure can be used to implement least recently used (LRU) page fault algorithm and Quick short algorithm.
(C) Queue data structure can be used to implement Quick short algorithm but not least recently used (LRU) page fault algorithm.

(D) Both (A) and (C)

35. Which of the following is not the type of queue?
(A) Ordinary queue
(B) Single ended queue
(C) Circular queue
(D) Priority queue

36. What does the following piece of code do where queue elements are integer?

```
void foo(){
if(isEmpty())
return -999;
else
{
int high;
high = q[front];
return high;
}
}
```

(A) Dequeue
(B) Enqueue
(C) Return the front element
(D) None

37. A Queue data structure serves major role in
(A) In recursive function calls
(B) Simulation of arbitrary linked list
(C) Simulation of limited resource allocation
(D) All of the above

38. What is the time complexity of enqueue operation?
(A) O(logn)
(B) O(nlogn)
(C) O(n)
(D) O(1)

39. What is a dequeue?
(A) A queue with insert/delete defined for both front and rear ends of the queue
(B) A queue implemented with a doubly linked list
(C) A queue implemented with both singly and doubly linked lists

(D) None of the mentioned

40. Consider a standard Circular Queue 'q' implementation (which has the same condition for Queue Full and Queue Empty) whose size is 17 and the elements of the queue are q[0], q[1], q[2].....,q[16]. The front and rear pointers are initialized to point at q[7] . In which position will the thirteenth element be added?
(A) q[16]
(B) q[2]
(C) q[3]
(D) q[4]

41. What are the applications of dequeue?
(A) A-Steal job scheduling algorithm
(B) Can be used as both stack and queue
(C) To find the maximum of all sub arrays of size k
(D) All of the above.

42. What is the time complexity of deleting from the rear end of the dequeue implemented with a singly linked list?
(A) $O(n \log n)$
(B) $O(\log n)$
(C) $O(n)$
(D) $O(n^{1.5})$

43. In linked list implementation of a queue, from where is the item deleted?
(A) At the head of link list
(B) At the centre position in the link list
(C) At the tail of the link list
(D) None of the mentioned

44. In linked list implementation of a queue, front and rear pointers are tracked. Which of these pointers will change during an insertion into EMPTY queue?
(A) Only front pointer
(B) Only rear pointer
(C) Both front and rear pointer
(D) None of the pointers

45. After performing these set of operations, what does the final list look contain?
InsertFront(10); InsertFront(20); InsertRear(30); DeleteFront(); InsertRear(40); InsertRear(10); DeleteRear(); InsertRear(15); display();
(A) 10 30 10 15

(B) 20 30 40 15
(C) 20 30 40 10
(D) 10 30 40 15

46. Which of the following is not an application of priority queue?
(A) Huffman codes
(B) Interrupt handling in operating system
(C) Undo operation in text editors
(D) Bayesian spam filter

47. What is the worst case time complexity to insert a node in a priority queue?
a) O(nlogn)
b) O(logn)
c) O(n)
d) O(n^2)

4 Linked List

1. What is the following function doing for a given singly Linked List with first node pointed by a pointer named as *head*?

```
void fun1(struct node* head){
if(head == NULL)
return;
fun1(head->next);
printf(”%d ”, head->data);
}
```

(A) Prints all nodes of linked lists
(B) Prints all nodes of linked list in reverse order
(C) Prints alternate nodes of Linked List
(D) Prints alternate nodes in reverse order

2. What is the following function doing for a given singly Linked List with first node pointed by a pointer named as *head*?

```
void fun1(struct node* head){
if(head == NULL)
return;
struct node* p=head;
while(p->next){
printf(”%d ”, p->next->data);
```