

CS301-P Data Structure Solved MCQ by Pin2.

. What is a data structure?

- a) A programming language**
- b) A collection of algorithms**
- c) A way to store and organize data**
- d) A type of computer hardware**

View Answer

Answer: c

Explanation: A data structure is a way to store and organize data efficiently, enhancing access and manipulation, unlike programming languages, algorithms, or computer hardware.

2. What are the disadvantages of arrays?

- a) Index value of an array can be negative**
- b) Elements are sequentially accessed**
- c) Data structure like queue or stack cannot be implemented**
- d) There are chances of wastage of memory space if elements inserted in an array are lesser than the allocated size**

View Answer

Answer: d

Explanation: Arrays are of fixed size. If we insert elements less than the allocated size, unoccupied positions can't be used again. Wastage will occur in memory.

3. Which data structure is used for implementing recursion?

- a) Stack**
- b) Queue**
- c) List**
- d) Array**

View Answer

Answer: a

Explanation: Stacks are used for the implementation of Recursion.

4. The data structure required to check whether an expression contains a balanced parenthesis is?

- a) Queue
- b) Stack
- c) Tree
- d) Array

View Answer

Answer: b

Explanation: The stack is a simple data structure in which elements are added and removed based on the LIFO principle. Open parenthesis is pushed into the stack and a closed parenthesis pops out elements till the top element of the stack is its corresponding open parenthesis. If the stack is empty, parenthesis is balanced otherwise it is unbalanced.

5. Which of the following is not the application of stack?

- a) Data Transfer between two asynchronous process
- b) Compiler Syntax Analyzer
- c) Tracking of local variables at run time
- d) A parentheses balancing program

View Answer

Answer: a

Explanation: Data transfer between the two asynchronous process uses the queue data structure for synchronisation. The rest are all stack applications

6. Which data structure is needed to convert infix notation to postfix notation?

- a) Tree
- b) Branch
- c) Stack
- d) Queue

View Answer

Answer: c

Explanation: The Stack data structure is used to convert infix expression to postfix expression. The purpose of stack is to reverse

the order of the operators in the expression. It also serves as a storage structure, as no operator can be printed until both of its operands have appeared.

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

- a) 74
- b) -18
- c) 22
- d) 40

View Answer

Answer: b

Explanation: Postfix Expression is $(6*(3-(2+4)))$ which results -18 as output.

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

- a) Stack
- b) Linked List
- c) Tree
- d) Queue

View Answer

Answer: a

Explanation: In recursive algorithms, the order in which the recursive process comes back is the reverse of the order in which it goes forward during execution. The compiler uses the stack data structure to implement recursion. In the forwarding phase, the values of local variables, parameters and the return address are pushed into the stack at each recursion level. In the backing-out phase, the stacked address is popped and used to execute the rest of the code.

9. Which of the following statement(s) about stack data structure is/are NOT correct?

- a) Top of the Stack always contain the new node
- b) Stack is the FIFO data structure
- c) Null link is present in the last node at the bottom of the stack
- d) Linked List are used for implementing Stacks

View Answer

Answer: b

Explanation: Stack follows LIFO.

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

- a) Array**
- b) Stack**
- c) Tree**
- d) Queue**

View Answer

Answer: d

Explanation: In Breadth First Search Traversal, BFS, starting vertex is first taken and adjacent vertices which are unvisited are also taken. Again, the first vertex which was added as an unvisited adjacent vertex list will be considered to add further unvisited vertices of the graph. To get the first unvisited vertex we need to follow First In First Out principle. Queue uses FIFO principle.

11. The prefix form of $A-B / (C * D ^ E)$ is?

- a) $-A/B * C ^ DE$**
- b) $-A/BC * ^ DE$**
- c) $-ABCD * ^ DE$**
- d) $- / * ^ ACBDE$**

View Answer

Answer: a

Explanation: Infix Expression is $(A-B)/(C * D ^ E)$

$(-A/B)(C * D ^ E)$

$-A/B * C ^ DE$. Thus prefix expression is $-A/B * C ^ DE$.

12. Which of the following points is/are not true about Linked List data structure when it is compared with an array?

- a) Random access is not allowed in a typical implementation of Linked Lists**

b) Access of elements in linked list takes less time than compared to arrays

c) Arrays have better cache locality that can make them better in terms of performance

d) It is easy to insert and delete elements in Linked List

View Answer

Answer: b

Explanation: To access an element in a linked list, we need to traverse every element until we reach the desired element. This will take more time than arrays as arrays provide random access to its elements.

13. Which data structure is based on the Last In First Out (LIFO) principle?

a) Tree

b) Linked List

c) Stack

d) Queue

View Answer

Answer: c

Explanation: The data structure that follows the Last In First Out (LIFO) principle is the Stack. It operates like a stack of objects, making it suitable for specific-order management.

14. Which of the following application makes use of a circular linked list?

a) Recursive function calls

b) Undo operation in a text editor

c) Implement Hash Tables

d) Allocating CPU to resources

View Answer

Answer: d

Explanation: Generally, round robin fashion is employed to allocate CPU time to resources which makes use of the circular linked list data structure. Recursive function calls use stack data structure. Undo

Operation in text editor uses doubly linked lists. Hash tables uses singly linked lists.

15. What is a bit array?

- a) Data structure that compactly stores bits
- b) Data structure for representing arrays of records
- c) Array in which elements are not present in continuous locations
- d) An array in which most of the elements have the same value

View Answer

Answer: a

Explanation: It compactly stores bits and exploits bit-level parallelism.

16. Which of the following tree data structures is not a balanced binary tree?

- a) Splay tree
- b) B-tree
- c) AVL tree
- d) Red-black tree

View Answer

Answer: b

Explanation: All the tree data structures given in options are balanced, but B-tree can have more than two children.

17. Which of the following is not the type of queue?

- a) Priority queue
- b) Circular queue
- c) Single ended queue
- d) Ordinary queue

View Answer

Answer: c

Explanation: Queue always has two ends. So, single ended queue is not the type of queue.

18. Which of the following data structures can be used for parentheses matching?

- a) n-ary tree

- b) queue
- c) priority queue
- d) stack

View Answer

Answer: d

Explanation: For every opening brace, push it into the stack, and for every closing brace, pop it off the stack. Do not take action for any other character. In the end, if the stack is empty, then the input has balanced parentheses.

19. Which algorithm is used in the top tree data structure?

- a) Backtracking
- b) Divide and Conquer
- c) Branch
- d) Greedy

View Answer

Answer: b

Explanation: Top tree is a type of data structure which is based on unrooted dynamic binary tree and is used to solve path related problems. It allows an algorithm called divide and conquer.

20. What is the need for a circular queue?

- a) easier computations
- b) implement LIFO principle in queues
- c) effective usage of memory
- d) to delete elements based on priority

View Answer

Answer: c

Explanation: In a linear queue, dequeue operation causes the starting elements of the array to be empty, and there is no way you can use that space, while in a circular queue, you can effectively use that space. Priority queue is used to delete the elements based on their priority. Higher priority elements will be deleted first whereas lower

priority elements will be deleted next. Queue data structure always follows FIFO principle.

21. Which of the following is the most widely used external memory data structure?

- a) B-tree
- b) Red-black tree
- c) AVL tree
- d) Both AVL tree and Red-black tree

View Answer

Answer: a

Explanation: In external memory, the data is transferred in form of blocks. These blocks have data valued and pointers. And B-tree can hold both the data values and pointers. So B-tree is used as an external memory data structure.

22. Which of the following is also known as Rope data structure?

- a) Linked List
- b) Array
- c) String
- d) Cord

View Answer

Answer: d

Explanation: Array is a linear data structure. Strings are a collection and sequence of codes, alphabets or characters. Linked List is a linear data structure having a node containing data input and the address of the next node. The cord is also known as the rope data structure.

23. What will be the output of the following program?

```

main()
{
    char str[]="san foundry";
    int len = strlen(str);
    int i;

    for(i=0;i<len;i++)
        push(str[i]); // pushes an element into si

    for(i=0;i<len;i++)
        pop(); //pops an element from the stack
}

```

- a) yrdnuof nas
 - b) foundry nas
 - c) sanfoundry
 - d) san foundry
- View Answer**

Answer: a

Explanation: First, the string 'san foundry' is pushed one by one into the stack.

When it is popped, the output will be as 'yrdnuof nas'.

24. Which of the following data structure can provide efficient searching of the elements?

- a) binary search tree
- b) unordered lists
- c) 2-3 tree
- d) treap

View Answer

Answer: c

Explanation: The average case time for lookup in a binary search tree, treap and 2-3 tree is $O(\log n)$ and in unordered lists it is $O(n)$. But in the worst case, only the 2-3 trees perform lookup efficiently as it takes $O(\log n)$, while others take $O(n)$.

25. What is an AVL tree?

- a) a tree which is unbalanced and is a height balanced tree
- b) a tree which is balanced and is a height balanced tree
- c) a tree with atmost 3 children
- d) a tree with three children

View Answer

Answer: b

Explanation: It is a self balancing tree with height difference atmost 1.

26. What is the time complexity for searching a key or integer in Van Emde Boas data structure?

- a) $O(M!)$**
- b) $O(\log M!)$**
- c) $O(\log(\log M))$**
- d) $O(M^2)$**

View Answer

Answer: c

Explanation: In order to search a key or integer in the Van Emde Boas data structure, the operation can be performed on an associative array. Hence, the time complexity for searching a key or integer in Van Emde Boas data structure is $O(\log(\log M))$.

27. The optimal data structure used to solve Tower of Hanoi is

-
- a) Tree**
 - b) Heap**
 - c) Priority queue**
 - d) Stack**

View Answer

Answer: d

Explanation: The Tower of Hanoi involves moving of disks 'stacked' at one peg to another peg with respect to the size constraint. It is conveniently done using stacks and priority queues. Stack approach is widely used to solve Tower of Hanoi.

28. What is the use of the bin data structure?

- a) to have efficient traversal**
- b) to have efficient region query**
- c) to have efficient deletion**
- d) to have efficient insertion**

View Answer

Answer: b

Explanation: Bin data structure allows us to have efficient region queries. A frequency of bin is increased by one each time a data point falls into a bin.

29. Which is the most appropriate data structure for reversing a word?

- a) stack
- b) queue
- c) graph
- d) tree

View Answer

Answer: a

Explanation: Stack is the most appropriate data structure for reversing a word because stack follows LIFO principle.

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

```
public void display()
{
    if(size == 0)
        System.out.println("underflow");
    else
    {
        Node current = first;
        while(current != null)
        {
            System.out.println(current);
            current = current.getNext();
        }
    }
}
```

- a) display the list
- b) reverse the list
- c) reverse the list excluding top-of-the-stack-element
- d) display the list excluding top-of-the-stack-element

View Answer

Answer: a

Explanation: An alias of the node 'first' is created which traverses through the list and displays the elements.

31. Which of the following is the simplest data structure that supports range searching?

- a) AA-trees**
- b) K-d trees**
- c) Heaps**
- d) binary search trees**

View Answer

Answer: c

Explanation: K-d trees are the simplest data structure that supports range searching and also it achieves the respectable running time.

32. What is the advantage of a hash table as a data structure?

- a) easy to implement**
- b) faster access of data**
- c) exhibit good locality of reference**
- d) very efficient for less number of entries**

View Answer

Answer: b

Explanation: Hash table is a data structure that has an advantage that it allows fast access of elements. Hash functions are used to determine the index of any input record in a hash table.

33. Which type of data structure is a ternary heap?

- a) Hash**
- b) Array**
- c) Priority Stack**
- d) Priority Queue**

View Answer

Answer: d

Explanation: Ternary heap is a type of data structure in the field of computer science. It is a part of the Heap data structure family. It is a priority queue type of data structure that follows all the property of heap.

34. What is a dequeue?

- a) A queue implemented with both singly and doubly linked lists
- b) A queue with insert/delete defined for front side of the queue
- c) A queue with insert/delete defined for both front and rear ends of the queue
- d) A queue implemented with a doubly linked list

View Answer

Answer: c

Explanation: A dequeue or a double ended queue is a queue with insert/delete defined for both front and rear ends of the queue.

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

- a) Priority queue
- b) Dequeue
- c) Circular queue
- d) Queue

View Answer

Answer: b

Explanation: In dequeuer, we can insert or delete elements from both the ends. In queue, we will follow first in first out principle for insertion and deletion of elements. Element with least priority will be deleted in a priority queue.

36. What is the output of the following Java code?

```
public class array
{
    public static void main(String args[])
    {
        int []arr = {1,2,3,4,5};
        System.out.println(arr[2]);
        System.out.println(arr[4]);
    }
}
```

- a) 4 and 2
- b) 2 and 4
- c) 5 and 3
- d) 3 and 5

View Answer

Answer: d

Explanation: Array indexing starts from 0.

37. In simple chaining, what data structure is appropriate?

- a) Doubly linked list**
- b) Circular linked list**
- c) Singly linked list**
- d) Binary trees**

View Answer

Answer: a

Explanation: Deletion becomes easier with doubly linked list, hence it is appropriate.