VIVA - VOCE

CHAPTER - 1

.1. **Define data type.**

ns. Data type is a method for declaring a set of variable of a particular type.

.2. **State the basic data types available in 'C' programming.**

ns. The basic data types available in 'C' programming are :

(i) int

(ii) float

(iii) char

(iv) double

.3. **What is derived data type?**

ns. Derived data type is derived from the basic data types.

Example : Arrays, Structure, Union.

.4. **What is data structure?**

ns. A data structure is a way of organizing data that considers not only the items stored, but also their relationship to each other.

.5. **What is primitive data structure?**

ns. Primitive data structure is the basic or built in type data structure which directly operated by machine instruction.

Example : Integer, float, character, double.

.6. **What is non primitive data structure?**

ns. Non primitive data structure is derived from the basic or primitive data structure.

Example : Array, list, files.

.7. **What is linear data structure?**

ns. The elements of a linear data structure are arranged in sequence in memory. The elements have a one to one relationship with other elements.

.8. **What is non linear data structure?**

ns. In non linear data structure, the elements have one to many relationship between them.

Example : Tree, graph.

Q.9. Define an algorithm.

Ans. An algorithm is a finite set of instructions written to accomplish a particular task.

Q.10. State the characteristics of an algorithm.

Ans. The characteristics of algorithm are as follows :

(i) Finiteness

(ii) Definiteness

(iii) Effectiveness

(iv) Input

(v) Output

Q.11. State the approaches to design an algorithm.

Ans. There are two approaches to design an algorithm, they are :

(i) Top-Down Approach.

(ii) Bottom-up Approach.

Q.12. What is time complexity?

Ans. The time complexity of an algorithm is the amount of time taken by the program for execution.

Q.13. What is space complexity?

Ans. The space complexity of an algorithm is the amount of memory needs to run to completion.

Q.14. What is Big 'O' Notation?

Ans. Big 'O' notation is used to describe the performance or complexity of an algorithm.

Q.15. How an algorithm can be analyzed?

Ans. Analysis of an algorithm is done on the following factors.

(i) Best case time complexity.

(ii) Worst case time complexity.

(iii) Average case time complexity.

CHAPTER - 2

Q.1. Define an array.

Ans. An array is a collection of similar data elements.

Q.2. What is sorting?

Ans. Sorting is a process of ordering a list of elements in either ascending or descending order.

Q.3. What are the types of sorting?

Ans. There are two types of sorting :

(a) Internal sorting (b) External sorting.

Q.4. What is internal sorting?

Ans. Internal sorting is done on data which is stored in the main memory.

Q.5. What is external sorting?

Ans. External sorting is done when data is stored in auxiliary or secondary storage devices.

Q.6. List some popular sorting methods.

Ans. Popular sorting methods are :

(1) Bubble sort.

(2) Insertion sort.

(3) Quick sort.

(4) Merge sort.

(5) Bucket sort.

(6) Selection sort.

(7) Shell sort.

Q.7. What is a bubble sort?

Ans.

- Bubble sort is a sorting technique which can be applied to data structures like arrays.
- Here, the adjacent values are compared and their positions are exchanged if they are out of order.
- The smaller value bubbles up to the top of the list, while the larger value sinks to the bottom.
- It is also called as comparison sort as it uses comparisons.

Q.8. What is insertion sort?

Ans.

- In insertion sort we pick up a particular value and then insert it at the appropriate place in the sorted sub list.
- This algorithm is very popular with bridge players when they sort their cards.

Q.9. What is quick sort?

Ans.

- Quick sort employs the 'divide and conquer' concept by dividing the list of elements into two sub elements.
- Select an element, pivot, from the list. Rearrange the elements in the list, so that all elements those are less than the pivot are arranged before the pivot and all elements those are greater than the pivot are arranged after the pivot. Now the pivot is in its position.
- Sort both sub lists of the elements which are less than the pivot and the list of elements which are more than the pivot recursively.

Q.10. What is merge sort?

Ans.

- In merge sort, the length of the list is 0 or 1, and then it is considered as sorted. Otherwise, divide the unsorted list into 2 lists each about half the size.
- Sort each sub list recursively. Implement the step 2 until the two sub lists are sorted. As a final step, combine (merge) both the lists back into one sorted list.

Q.11. What is selection sort?

Ans.

- In selection sort, the first pass, we find the smallest element from a[0], a[1], a[2], ... a[n–1] and swap it with the first element, i.e. a[0].
- In the second pass, we find the smallest element from a[1], a[2], a[3]....a[n–1] and swap it with a[1] and so on.

Q.12. Give the comparisons of sorting algorithms on the different parameters.

Ans.

SORT	BEST	AVERAGE	WORST	MEMORY SPACE	STABILITY	METHOD
BUBBLE	O(n)	$O(n^2)$	$O(n^2)$	O(1) constant	Stable	Exchange
SELECTION	O(n)	$O(n^2)$	$O(n^2)$	O(1) constant	Stable	Selection

SORT	BEST	AVERAGE	WORST	MEMORY SPACE	STABILITY	METHOD
INSERTION	O(n)	$O(n^2)$	$O(n^2)$	O(1) constant	Stable	Insertion
QUICK	O(n)	O(n log n)	$O(n^2)$	O(1) constant	Stable	Partitioning
HEAP	O(n log n)	O(n log n)	O(n log n)	O(1) constant	Instable	Selection
MERGE	O(n log n)	O(n log n)	O(n log n)	Depends	Instable	Merging
SHELL	O(n)	$O(n \log n^2)$	O(n log n)	O(1) constant	Instable	Insertion
RADIX	O(n log n)	O(n log n)	$O(n^2)$	Depends	Stable	Partitioning

Q.13. What is searching?

Ans. Searching is an operation which finds the place of given elements in the list.

Q.14. What are the types of searching?

Ans. There are two types of searching:
(i) Linear search (ii) Binary search

Q.15. State the principle of linear search.

Ans. In linear search, the searching begins from the first record. Searching continues sequentially till the record with a matching key value is found.

Q.16. State the principle of binary search.

Ans. In binary search, the records are partitioned into two parts and the required key is compared with the middle record. If a record is not found in the middle, then search proceeds in the upper or lower half. Search is successful when the record is found.

CHAPTER - 3

Q.1. What is stack?

Ans. A stack is a linear data structure in which elements are added and removed only from one end called top of stack.

Q.2. State the operations performed on a stack.

Ans. The operations performed on stacks are :
(i) Push operation (ii) Pop operation
(iii) Peep operation.

Q.3. What is the significance of top?

Ans. The elements in a stack are added and removed only from one end called Top of stack (Top).

Q.4. When stack memory is overflow?

Ans. If TOP=MAX–1, then the stack is overflow and no more insertions can be further done.

Q.5. When stack memory is underflow?

Ans. If TOP=NULL, then it means the stack is empty and no more deletions can be further done.

Q.6. What is peep operation?

Ans. Peep is an operation that returns the value of the topmost element of the stack without deleting it from the stack.

Q.7. What is push operation?

Ans. The push operation is used to insert an element into the stack.

Q.8. What is pop operation?

Ans. The pop operation is used to delete the topmost element from the stack.

Q.9. What is polish notation?

Ans. In polish notations, the operator is written before its operand or after its operand.

Q.10. Define infix notation.

Ans. In infix notation, operator is placed in between the operands.

Q.11. Define prefix notation.

Ans. In prefix notation, operator is placed before its operands.

Q.12. Define postfix notation?

Ans. In postfix notation operator is placed after its operands.

Q.13. What is recursion?

Ans. A function which calls itself is called as recursion.

Q.14. State the applications of stack.

Ans. The applications of stack are as follows :

(i) To evaluate recursive functions.

(ii) To evaluate arithmetic expressions.

CHAPTER - 4

Q.1. What is queue?

Ans. Queue is a linear data structure. Insertion of an item is done at rear end. Deletion of an item is done at front end.

Q.2. What are the types of queue?

Ans. The types of queues are :

(i) Linear queue.

(ii) Circular queue.

(ii) Double ended queue.

(iv) Priority queue.

Q.3. What is queue overflow?

Ans. If rear = MAX – 1, then queue overflow occurs where MAX is the maximum size of queue.

Q.4. What is queue underflow?

Ans. When queue is totally empty i.e. no element is present in the queue. This condition is called as queue underflow.

Q.5. What is circular queue?

Ans. A circular queue is one in which the first element comes just after the last element.

Q.6. What is the advantages of circular queue over linear queue?

Ans. Circular queue have less memory consumption as compared to linear queue.

Q.7. Define dequeues.

Ans. In dequeue, the elements can be inserted or deleted at both the ends.

Q.8. What is input restricted dequeue?

Ans. In this dequeue, insertions can be done only at one end while deletions can be done from both ends.

Q.9. What is output restricted dequeue?

Ans. In this dequeue, deletions can be done only at one end while insertions can be done on both ends.

Q.10. What is priority queue?

Ans. A priority queue in which each element is assigned a priority. An element with higher priority is processed before an element with a lower priority.

Q.11. What are the advantages of priority queue?

Ans. It is useful for prioritizing jobs in operating system. It is also used in simulations.

Q.12. What are the applications of queues?

Ans. The applications of queues are :

(i) Multilevel queue scheduling.

(ii) Simulation.

CHAPTER - 5

Q.1. What is linked list?

Ans. Linked list is a collection of data elements where one element is connected to another element by using pointer.

Q.2. State the limitations of an array over the link list.

Ans. The limitations of an array are :

(i) Array size remains fixed.

(ii) During run time space cannot be increased.

(iii) Insertion and deletion of middle elements take more time.

Q.3. What is the significance of NULL pointer?

Ans. The NULL pointer indicates the end of the list.

Q.4. State the types of linked list?

Ans. The various types of linked list are :

(i) Singly linked list.

(ii) Doubly linked list.

(iii) Circular linked list.

(iv) Circular doubly linked list.

Q.5. What is the use of malloc function?

Ans. Malloc function allocates memory during run time.

Q.6. What is singly linked list?

Ans. In singly linked list, each node contains only one pointer which points to the next node of the list.

Q.7. What is doubly linked list?

Ans. In doubly linked list, each node contains two pointers, one pointing to the previous node and the other pointing to the next node.

Q.8. What is circular list?

Ans. In circular linked list, the last node does not contain a NULL pointer, the last node points back to the first node.

Q.9. State the operations performed on a linked list.

Ans. The operations performed on a linked list are :

(a) Creation.

(b) Traversing the list.

(c) Insertion.

(d) Deletion.

(e) Searching.

(f) Reversing.

(g) Concatenation.

Q.10. What are the advantages of linked list?

Ans. The advantages of linked list are as follows :

(i) Insertion and deletion of elements is easy and efficient.

(ii) There is no memory wastage.

(iii) With large records, moving pointers is easier and faster than moving the items themselves.

(iv) They can grow or shrink during the execution of a program.

Q.11. What are the limitation of linked lists?

Ans. The limitation of linked list are :

(i) Linked list consumes extra space as compared to an array.

(ii) Linked list do not allow random access.

(iii) Programming is difficult with pointers.

Q.12. What are the advantages of circular list?

Ans. The advantages of circular list are :

(i) Each node can be accessed from any node where as a singly list can be accessed only by starting from the nodes.

(ii) No need of NULL pointer as all pointers contains valid address.

Q.13. What are the disadvantages of circular linked list?

Ans. Disadvantage of circular linked list is :

An infinite looping can be caused while traversing it if proper care is not taken. This can be avoided by replacing reader node at the beginning of circular linked list.

CHAPTER - 6

Q.1. Define tree.

Ans. A tree is defined as a finite set of one or more nodes. Trees are mainly used to represent data containing a hierarchical relationship between the elements.

Q.2. What is the degree of a node?

Ans. Degree of a node is equal to the number of children that a node has.

Q.3. Define the terms leaf node and siblings.

Ans.

(i) **Leaf node :** A node that has no children is called the leaf node.

(ii) **Siblings :** Children of the same parent node are called sibling node.

Q.4. What is path?

Ans. A sequence of consecutive edges is called a path.

Q.5. What is depth / Height of a tree?

Ans. The length of the path from the root node to the deepest node in the tree,is known as the height of the tree.

Q.6. Define binary tree.

Ans. Binary tree is a special type of tree in which every node or vertex has either no children, one or two children.

Q.7. What is a complete binary tree?

Ans. In a complete binary tree, every level except possibly the last is completely filled.

Q.8. What is binary search tree?

Ans. In a binary search tree, all the nodes in the left sub-tree has a value less than that of the root node.

Q.9. Construct a binary search tree for the following values.

50, 33, 44, 22, 77, 35, 60, 40

Ans.

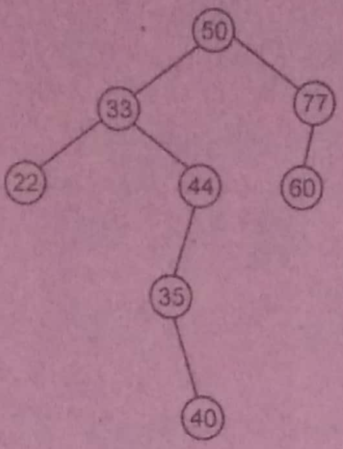

Fig.

Q.10. What is traversal of a tree?

Ans. Traversing is the process of visiting each node in the tree exactly once.

Q.11. State the methods of binary tree traversal.

Ans. The methods of binary tree traversal are :

(i) Preorder Traversal.

(ii) Inorder Traversal.

(iii) Postorder Traversal.

Q.12. State the principle of preorder traversal.

Ans. In preorder traversal following steps are followed :

(i) Visiting the root node

(ii) Traversing the left subtree

(iii) Traversing the right subtree.

Q.13. State the principle of inorder traversal.

Ans. The principle of inorder traversal is :

(i) Traversing the left subtree.

(ii) Visiting the root node

(iii) Finally traversing the right subtree.

Q.14. State the principle of postorder traversal.

Ans. The principle of postorder traversal is :

(i) Traversing the left subtree

(ii) Visiting the root node

(iii) Finally traversing the right subtree.

Q.15. What is an expression tree?

Ans. A binary tree that is used to evaluate expressions is known as an expression tree.

Q.16. Draw the tree structure of the expression Z × (A + B) × C / D.

Ans.

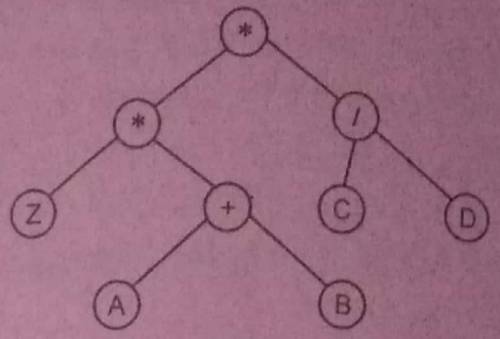

Fig.

Q.17. What is skewed binary tree?

Ans. A skewed binary tree could be skewed to the left or it could be skewed to the right.

Q.18. What is extended binary tree or square tree?

Ans. An extended binary tree is a tree in which each empty subtree is replaced by an external node. It is denoted by using square (□).

CHAPTER - 7

Q.1. What is graph?

Ans. A graph is a collection of two sets V and E where V is the set of vertices and E is the set of edges.

Q.2. Define degree of node or vertex.

Ans. The degree of a vertex is the number of edges incident to it.

Q.3. Define in-degree of vertex.

Ans. The in-degree of a vertex V is the number of edges that have V as the head.

Q.4. Define out-degree of vertex.

Ans. The out-degree of vertex V is the number of edges that have V as the tail.

Q.5. What is weighted graph?

Ans. A weighted graph is a graph in which weights are assigned to edges.

Q.6. What is directed graph?

Ans. A directed graph in which each edge is assigned a direction.

Q.7. What is complete graph?

Ans. A graph G is said to be complete graph if there exists an edge between every pair of vertices.

Q.8. What is connected graph?

Ans. A graph G is said to be connected if there is at least one path between every pair of vertices.

Q.9. What is an adjacency matrix?

Ans. An adjacency matrix contains only 0's and 1's is called an adjacency matrix.

Q.10. State two methods of graph traversal.

Ans. The two methods of graph traversal are :

(i) Depth-first search.

(ii) Breadth first search.

Q.11. What is hashing?

Ans. Hashing is a search technique used for database indexing.

Q.12. Define hash function.

Ans. Hash function generates a single integer that may serve as an index to an array.

Q.13. What is symbol table?

Ans. The table used by compiler to maintain information about symbols from a program.

Q.14. State different methods to evaluate hash functions.

Ans. The different methods to evaluate the hash functions are as follows :

(i) Mid square method.

(ii) Folding method.

(iii) Division method.

(iv) Digit analysis.

(v) Length dependent method.

Q.15. What is a collision in hashing?

Ans. When the hash function generates the same integer on different keys then it results into a collision.

Q.16. What is collision resolution techniques?

Ans. A method used to solve the problem of collision is called the collision resolution techniques.

Q.17. What are the different collision resolution methods?

Ans. There are two methods for collision resolutions :

(i) Open addressing.

(ii) Chaining.

Q.18. What is linear probing.

Ans. Linear probing is a collision resolution technique. It resolves the collision by simply checking the next slot.

Q.19. What is rehashing?

Ans. If an overflow occurs, a new address is computed by using another hash function.

Q.20. What is chaining?

Ans. In chaining two tables are maintained in memory, in which one table contains all the records in memory and second table contains pointer to the linked list.