QUESTIONS ON C PROGRAM AND DATA STRUCTURES

1) What is difference between i++ and ++i?

1) The expression 'i++' returns the old value and then increments i. The expression ++i increments the value and returns new value.
2) Precedence of postfix ++ is higher than that of prefix ++.
3) Associativity of postfix ++ is left to right and associativity of prefix ++ is right to left.
4) In C++, ++i can be used as l-value, but i++ cannot be. In C, they both cannot be used as l-value.

2) What Does Static Variable Mean?

Static variable is available to a C application, throughout the life time. At the time of starting the program execution, static variables allocations takes place first. In a scenario where one variable is to be used by all the functions (which is accessed by main () function), then the variable need to be declared as static in a C program.

3) What Is A Null Pointer?

A null pointer is a special pointer value that is known not to point anywhere. It means that no other valid pointer, to any other variable or array cell or anything else, will ever compare equal to a null pointer.

4) What Is The Difference Between Call By Value And Call By Reference?

When using Call by Value, you are sending the value of a variable as parameter to a function, whereas Call by Reference sends the address of the variable. Also, under Call by Value, the value in the parameter is not affected by whatever operation that takes place, while in the case of Call by Reference, values can be affected by the process within the function.

5) What Are Header Files And What Are Its Uses In C Programming?

Header files are also known as library files. They contain two essential things: the definitions and prototypes of functions being used in a program. Simply put, commands that you use in C programming are actually functions that are defined from within each header files. Each header file contains a set of functions. For example: stdio.h is a header file that contains definition and prototypes of commands like printf and scanf.

6) Differentiate Source Codes from Object Codes

Source codes are codes that were written by the programmer. It is made up of the commands and other English-like keywords that are supposed to instruct the computer what to do. However, computers would not be able to understand source codes. Therefore, source codes are compiled using a compiler. The resulting outputs are object codes, which are in a format that can be understood by the computer processor. In C programming, source codes are saved with the file extension .C, while object codes are saved with the file extension .OBJ

7) In C programming, how do you insert quote characters (' and ") into the output screen?

This is a common problem for beginners because quotes are normally part of a printf statement. To insert the quote character as part of the output, use the format specifiers \' (for single quote), and \" (for double quote).

8) Why is it that not all header files are declared in every C program?

The choice of declaring a header file at the top of each C program would depend on what commands/functions you will be using in that program. Since each header file contains different function definitions and prototype, you would be using only those header files that would contain the functions you will need. Declaring all header files in every program would only increase the overall file size and load of the program, and is not considered a good programming style.

9) When is the "void" keyword used in a function?

When declaring functions, you will decide whether that function would be returning a value or not. If that function will not return a value, such as when the purpose of a function is to display some outputs on the screen, then "void" is to be placed at the leftmost part of the function header. When a return value is expected after the function execution, the data type of the return value is placed instead of "void".

10) What is wrong in this statement? scanf("%d",whatnumber);

An ampersand & symbol must be placed before the variable name whatnumber. Placing & means whatever integer value is entered by the user is stored at the "address" of the variable name. This is a common mistake for programmers, often leading to logical errors.

11) What are preprocessor directives?

Preprocessor directives are placed at the beginning of every C program. This is where library files are specified, which would depend on what functions are to be used in the program. Another use of preprocessor directives is the declaration of constants.Preprocessor directives begin with the # symbol.

12) Describe the order of precedence with regards to operators in C.

Order of precedence determines which operation must first take place in an operation statement or conditional statement. On the top most level of precedence are the unary operators !, +, - and &. It is followed by the regular mathematical operators (*, / and modulus % first, followed by + and -). Next in line are the relational operators <, <=, >= and >. This is then followed by the two equality operators == and !=. The logical operators && and || are next evaluated. On the last level is the assignment operator =.

13) Why is C language being considered a middle level language?

This is because C language is rich in features that make it behave like a high level language while at the same time can interact with hardware using low level methods. The use of a well

structured approach to programming, coupled with English-like words used in functions, makes it act as a high level language. On the other hand, C can directly access memory structures similar to assembly language routines.

14) What are structure types in C?

Structure types are primarily used to store records. A record is made up of related fields. This makes it easier to organize a group of related data.

15) What are pointers?

Pointers point to specific areas in the memory. Pointers contain the address of a variable, which in turn may contain a value or even an address to another memory.

16). Can you pass an entire structure to functions?

Yes, it is possible to pass an entire structure to a function in a call by method style. However, some programmers prefer declaring the structure globally, then pass a variable of that structure type to a function. This method helps maintain consistency and uniformity in terms of argument type.

17) Are comments included during the compilation stage and placed in the EXE file as well?

No, comments that were encountered by the compiler are disregarded. Comments are mostly for the guidance of the programmer only and do not have any other significant use in the program functionality.

18) Difference between a stable and unstable sorting algorithm?

A sorting algorithm is said to be **stable** if it maintains the relative order of numbers/records in the case of tie i.e. if you need to sort 1 1 2 3 then if you don't change order of those first two ones than your algorithm is stable, but if you swap them then it becomes unstable, despite the overall result or sorting order remain same.

19) When does the worst case of QuickSort occur?

In quick sort, we select a pivot element, then partition the given array around the pivot element by placing pivot element at its correct position in sorted array.
The worst case of quickSort occurs when one part after partition contains all elements and other part is empty. For example, if the input array is sorted and if last or first element is chosen as a pivot, then the worst occurs.

20) How do Insertion sort work?

Insertion sort takes elements of the array sequentially, and maintains a sorted subarray to the left of the current point. It does this by taking an element, finding its correct position in the sorted array, and shifting all following elements by 1, leaving a space for the element to be inserted.

21) What are the key advantages of Insertion Sort?

It is efficient at sorting extremely short arrays due to a very low constant factor in its complexity. It is also extremely good at sorting arrays that are already "almost" sorted. A common use is for re-sorting arrays that have had some small updates to their elements.

22) Discuss best, average, and worst case time and memory complexity.

Insertion sort has an average and worst runtime of O(n^2), and a best runtime of O(n). It doesn't need any extra buffer, so space complexity is O(1).

23) How do Heapsort work?

Heapsort starts by building a max heap. A binary max heap is a nearly complete binary tree in which each parent node is larger or equal to its children. The heap is stored in the same memory in which the original array elements are. Once the heap is formed, it completely replaces the array. After that, we take and remove the first element, restore the heap property, thus reducing the heap size by 1, after which we place the max element at the end of that memory. This is repeated until we empty out the heap, resulting in the smallest element being in the first place, and the following elements being sequentially larger.

24) What is binary search?

Binary search is most useful when the list is sorted. In binary search, element present in the middle of the list is determined. If the key (number to search) is smaller than the middle element, the binary search is done on the first half. If the key (number to search) is greater than the middle element, the binary search is done on the second half (right). The first and the last half are again divided into two by determining the middle element.

25) Explain whether it is possible to use binary search for linked lists?

Since random access is not acceptable in linked list, it is impossible to reach the middle element of O(1) time. Thus, binary search is not possible for linked list.

26) What do you mean by an Array?

- Array is a set of similar data type.
- Arrays objects store multiple variables with the same type.
- It can hold primitive types and object references.
- Arrays are always fixed

27) Advantages and disadvantages of Array?

Advantages:

- We can put in place other data structures like stacks, queues, linked lists, trees, graphs, etc. in Array.
- Arrays can sort multiple elements at a time.
- We can access an element of Array by using an index.

Disadvantages:

- We have to declare Size of an array in advance. However, we may not know what size we need at the time of array declaration.
- The array is static structure. It means array size is always fixed, so we cannot increase or decrease memory allocation.

28) Can we change the size of an array at run time?

No we cannot change the array size. Though there are similar data types available which allow a change in size.

29) What will happen if in a C program you assign a value to an array element whose subscript exceeds the size of array?

The program may crash if some important data gets overwritten.

If the index of the array size is exceeded, the program will crash.But the modern compilers will take care of this kind of errors.

30) What does the following declaration mean?

int (*ptr)[10];

ptr is a pointer to an array of 10 integer.

In C, if you pass an array as an argument to a function, what actually gets passed?

When we pass an array as a funtion argument, the base address of the array will be passed.

31)What do you understand by Infix, Prefix, and Postfix notations?

Infix Notation – Operators are written between the operands. This is the standard way of writing expressions. For example, A * (B + C) / D

Postfix Notation – Operators are written after the operands, hence the name. For instance, A B C + * D /

Prefix Notation – Operators are written before the operands. / * A + B C D is the prefix notation equivalent of the aforementioned postfix notation example

32) What is Stack and where it can be used?

Stack is a linear data structure which the order LIFO(Last In First Out) or FILO(First In Last Out) for accessing elements. Basic operations of stack are :Push,Pop ,Peek.

33) What is LIFO?

LIFO is a short form of Last In First Out. It refers how data is accessed, stored and retrieved. Using this scheme, data that was stored last should be the one to be extracted first. This also means that in order to gain access to the first data, all the other data that was stored before this first data must first be retrieved and extracted.

34) What is the prefix and post fix notation of (a + b) * (c + d) ?

Prefix Notation − * + a b + c d

Postfix Notation − a b + c d + *

35) Which data structures are applied when dealing with a recursive function?

Recursion, is a function that calls itself based on a terminating condition, makes use of the stack. Using LIFO, a call to a recursive function saves the return address so that it knows how to return to the calling function after the call terminates.

36) What are multidimensional arrays?

Multidimensional arrays make use of multiple indexes to store data. It is useful when storing data that cannot be represented using single dimensional indexing, such as data representation in a board game, tables with data stored in more than one column.

37) What is FIFO?

FIFO stands for First-in, First-out, and is used to represent how data is accessed in a queue. Data has been inserted into the queue list the longest is the one that is removed first.

38) Differentiate NULL and VOID

Null is a value, whereas Void is a data type identifier. A variable that is given a Null value indicates an empty value. The void is used to identify pointers as having no initial size.

39) What is a postfix expression?

A postfix expression is an expression in which each operator follows its operands. The advantage of this form is that there is no need to group sub-expressions in parentheses or to consider operator precedence.

40) Which sorting algorithm is considered the fastest?

There are many types of sorting algorithms: quick sort, bubble sort, balloon sort, radix sort, merge sort, etc. Not one can be considered the fastest because each algorithm is designed for a particular data structure and data set. It would depend on the data set that you would want to sort.

41) Differentiate STACK from ARRAY.

Stack follows a LIFO pattern. It means that data access follows a sequence wherein the last data to be stored when the first one to be extracted. Arrays, on the other hand, does not follow a particular order and instead can be accessed by referring to the indexed element within the array.

42) Sorting Is Not Possible By Using Which Of The Following Methods? (insertion, Selection, Exchange, Deletion)

Sorting is not possible in Deletion. Using insertion we can perform insertion sort, using selection we can perform selection sort, using exchange we can perform the bubble sort (and other similar sorting methods). But no sorting method can be done just using deletion.

43) A normal queue, if implemented using an array of size MAX_SIZE, gets full when?

When Rear = MAX_SIZE – 1, there will be no space left for the elements to be added in queue. Thus queue becomes full.

44) The data structure required to check whether an expression contains balanced parenthesis is?

Stack can be used to check whether the expression contains balanced paranthesis or not.

45) What Is Sequential Search?

In sequential search each item in the array is compared with the item being searched until a match occurs. It is applicable to a table organized either as an array or as a linked list.

46) What Do You Mean By Overflow And Underflow?

When new data is to be inserted into the data structure but there is no available space i.e. free storage list is empty this situation is called overflow.

When we want to delete data from a data structure that is empty this situation is called underflow.

47) What is the difference between Structure and Union in C?

STRUCTURE	UNION
The keyword **struct** is used to define a structure	The keyword **union** is used to define a union.
When a variable is associated with a structure, the compiler allocates the memory for each member. The size of structure is **greater than or equal to the sum of sizes of its members.**	when a variable is associated with a union, the compiler allocates the memory by considering the size of the largest memory. So, size of **union is equal to the size of largest member.**
Each member within a structure is assigned unique storage area of location.	Memory allocated is shared by individual members of union.
Altering the value of a member will not affect other members of the structure.	Altering the value of any of the member will alter other member values.
Individual member can be accessed at a time.	Only one member can be accessed at a time.
Several members of a structure can initialize at once.	Only the first member of a union can be initialized.

48) What Is Dangling Pointer And How To Avoid It?

After a call to free(p) makes a subsequent reference to *p illegal, i.e. though the storage to p is freed but the value of p(address) remain unchanged .so the object at that address may be used as the value of *p (i.e. there is no way to detect the illegality).Here p is called dangling pointer.

To avoid this it is better to set p to NULL after executing free(p).The null pointer value doesn't reference a storage location it is a pointer that doesn't point to anything.

49) Difference Between Calloc And Malloc ?

malloc: allocate n bytes.
calloc: allocate m times n bytes initialized to 0.

50) How Is The Front Of The Queue Calculated ?

The front of the queue is calculated by front = (front+1) % size.

51) Why Do We Use A Multidimensional Array?

A multidimensional array can be useful to organize subgroups of data within an array. In addition to organizing data stored in elements of an array, a multidimensional array can store memory addresses of data in a pointer array and an array of pointers.

Multidimensional arrays are used to store information in a matrix form.
e.g; a railway timetable, schedule cannot be stored as a single dimensional array. One can use a 3-D array for storing height, width and length of each room on each floor of a building.

52) Is Pointer A Variable?

Yes, a pointer is a variable and can be used as an element of a structure and as an attribute of a class in some programming languages such as C++, but not Java. However, the contents of a pointer is a memory address of another location of memory, which is usually the memory address of another variable, element of a structure, or attribute of a class.

53) State The Advantages Of Using Postfix Notations?

- Need not worry about the rules of precedence.
- Need not worry about the rules for right to left associativity.
- Need not need parenthesis to override the above rules.

54) What are linear and non linear data Structures?

- **Linear:**A data structure is said to be linear if its elements form a sequence or a linear list. Examples: Array. Linked List, Stacks and Queues
- **Non-Linear:**A data structure is said to be non-linear if traversal of nodes is nonlinear in nature. Example: Graph and Trees.

55) Discuss how to implement queue using stack.

A queue can be implemented by using 2 stacks:-
1. An element is inserted in the queue by pushing it into stack 1
2. An element is extracted from the queue by popping it from the stack 2
3. If the stack 2 is empty then all elements currently in stack 1 are transferred to stack 2 but in the reverse order
4. If the stack 2 is not empty just pop the value from stack 2.

56) Is Using Exit () The Same As Using Return?

No, The exit () function is used to exit your program and return control to the operating system.The return statement is used to return from a function and return control to the calling function. If you issue a return from the main () function, you are essentially returning control to the calling function, which is the operating system. In this case, the return statement and exit () function are similar.

57) What Is A Nested Loop?

A nested loop is a loop that runs within another loop. Put it in another sense, you have an inner loop that is inside an outer loop. In this scenario, the inner loop is performed a number of times as specified by the outer loop. For each turn on the outer loop, the inner loop is first performed.

58) What Actions Are Performed When A Function Is Called?

When a function is called

i) arguments are passed.
ii) local variables are allocated and initialized.
ii) transferring control to the function.

59) What Actions Are Performed When A Function Returns?

i) Return address is retrieved.
ii) Function's data area is freed.
iii) Branch is taken to the return address.

60) What Method Removes The Value From The Top Of A Stack?

The pop() member method removes the value from the top of a stack, which is then returned by the pop() member method to the statement that calls the pop() member method.

61) **What Method Is Used To Place A Value Onto The Top Of A Stack?**

push() method, Push is the direction that data is being added to the stack. push() member method places a value onto the top of a stack.

62) When Can You Tell That A Memory Leak Will Occur?

A memory leak occurs when a program loses the ability to free a block of dynamically allocated memory.

63) State The Rules To Be Followed During Infix To Prefix Conversions?

- Fully parenthesize the expression starting from left to right. During parenthesizing, the operators having higher precedence are first parenthesized.
- Move the operators one by one to their left, such that each operator replaces their corresponding left parenthesis.
- The part of the expression, which has been converted into prefix is to be treated as single operand.
- Once the expression is converted into prefix form, remove all parenthesis.

64) State The Rules To Be Followed During Infix To Postfix Conversions?

- Fully parenthesize the expression starting from left to right. During parenthesizing, the operators having higher precedence are first parenthesized.
- Move the operators one by one to their right, such that each operator replaces their corresponding right parenthesis.
- The part of the expression, which has been converted into postfix is to be treated as single operand.
- Once the expression is converted into postfix form, remove all parenthesis.

65) Mention The Advantages Of Representing Stacks Using Linked Lists Than Arrays?

- It is not necessary to specify the number of elements to be stored in a stack during its declaration, since memory is allocated dynamically at run time when an element is added to the stack.
- Insertions and deletions can be handled easily and efficiently.
- Linked list representation of stacks can grow and shrink in size without wasting memory space, depending upon the insertion and deletion that occurs in the list.
- Multiple stacks can be represented efficiently using a chain for each stack.

66) State The Difference Between Primitive And Non-primitive Data Types?

Primitive data types are the fundamental data types. Eg) int, float, double, char Non-primitive data types are user defined data types. Eg) Structure, Union and enumerated data types.

67) Define An Abstract Data Type (adt)?

An abstract data type is a set of operations. ADTs are mathematical abstractions; now here in an ADT's definition is there any mention of how the set of operations is implemented. Objects such as lists, sets and graphs, along with their operations can be viewed as abstract data types.

***68).* How does insertion sort differ from selection sort?**

Both insertion and selection approaches maintain two sub-lists, sorted and unsorted. Each takes one element from the unsorted sub-list and place it into the sorted sub-list. The distinction between the two sorting processes lies in the treatment of the current element.

Insertion sort takes the current element and places it in the sorted sublist at the appropriate location. Selection sort, on the other hand, searches for the minimum value in the unsorted sub-list and replaces the same with the present element.

69) What are the issues that hamper the efficiency in sorting a file?

The issues are:

- Length of time required by the programmer in coding a particular sorting program.
- Amount of machine time necessary for running the particular program.
- The amount of space necessary for the particular program.

70) **What do you mean by the Scope of the variable? What is the scope of the variables in C?**

Scope of the variable can be defined as the part of the code area where the variables declared in the program can be accessed directly. In C, all identifiers are lexically (or statically) scoped.

71) Can I use int datatype to store 32768 value in C program?

No, Integer datatype will support the range between **-32768 and 32767.**Any value exceeding that will not be stored. We can either use **float** or **long int.**

72) Can I create a customized Header File in C language?

It is possible to create a new header file. Create a file with function prototypes that need to be used in the program. Include the file in the '#include' section in its name.

73) **What is typecasting?**

Typecasting is a process of converting one data type into another is known as typecasting. If we want to store the floating type value to an int type, then we will convert the data type into another data type explicitly.

74) Can we apply Binary search algorithm to a sorted Linked list?

No, we cannot apply the binary search algorithm to a sorted linked list because finding the index of the middle element is difficult.

75) Name the data structures using which Queue can be implemented?

Queue can be implemented using

- Arrays
- Linked List
- Stack

76) What is Singly Linked Lists

In a singly linked list, each node stores a reference to an object that is an element of the sequence, as well as a reference to the next node of the list. It does not store any pointer or reference to the previous node. To store a single linked list, only the reference or pointer to the first node in that list must be stored. The last node in a single linked list points to nothing.

77) Explain Doubly Linked Lists

A doubly linked list is a data structure that consists of a set of sequentially **l**inked records called nodes. Each node contains three fields: two link fields (references to the previous and to the next node in the sequence of nodes) and one data field.

78) List the basic operations carried out in a linked list?

The basic operations carried out in a linked list include:

- Creation of a list

• Insertion of a node

• Deletion of a node

• Modification of a node

• Traversal of the list

79) Explain Circular Linked List?

Circular Linked List is a variation of Linked list in which the first element points to the last element and the last element points to the first element. Both Singly Linked List and Doubly Linked List can be made into a circular linked list.

80) State the difference between arrays and linked lists?

Arrays	Linked Lists
Size of an array is fixed	Size of a list is variable
It is necessary to specify the number of elements during declaration.	It is not necessary to specify the number of elements during declaration
Insertions and deletions are somewhat difficult	Insertions and deletions are carried out easily
It occupies less memory than a linked list for the same number of elements	It occupies more memory

81) **List out the disadvantages of using a linked list**

• Searching a particular element in a list is difficult and time consuming

• A linked list will use more storage space than an array to store the same number of elements

82) What is Binary Tree?

A binary tree is a finite set of nodes which is either empty or consists of a root and two disjoint binary trees called the left sub-tree and right sub-tree.

83) **Define Degree of a Node**

The total number of sub-trees attached to that node is called the degree of the node. If the degree is zero, it is called a terminal or leaf node of a tree.

84) Define height and level of a node

- Height is the length of the longest path to a leaf.
- The level of a node n is the number of edges on the path from the root node to n.

85) Define a full binary tree

A full binary tree is a tree in which all the leaves are on the same level and every non-leaf node has exactly two children. A full binary tree is a tree in which every node other than the leaves has two children.

86) What are the tasks performed during inorder traversal?

- Traverse the left sub-tree
- Process the root node
- Traverse the right sub-tree

87) What are the tasks performed during postorder traversal?

- Traverse the left sub-tree
- Traverse the right sub-tree
- Process the root node

88) What are the tasks performed during preorder traversal?

- Process the root
- Traverse the left sub-tree
- Traverse the right sub-tree

89) Define a binary search tree?

A binary search tree is a special binary tree, which is either empty or it should satisfy the following characteristics:

Every node has a value and no two nodes should have the same value (i.e) the values in the binary search tree are distinct

- The values in any left sub-tree is less than the value of its parent node
- The values in any right sub-tree is greater than the value of its parent node
- The left and right sub-trees of each node are again binary search trees

90) Define AVL Trees?

AVL tree is a self-balancing Binary Search Tree (BST) where the difference between heights of left and right subtrees cannot be more than one for all nodes.

91) What do you mean by balance factor of a node in AVL tree?

The height of left subtree minus height of right subtree is called balance factor of a node in AVL tree. The balance factor may be either 0 or +1 or -1. The height of an empty tree is -1

92) What is meant by inorder successor of a Node in Binary Tree? For which node, Inorder Successor is NULL?

- Inorder successor of a node is the next node in Inorder traversal of the Binary Tree.
- Inorder Successor is NULL for the last node in Inoorder traversal.

93) What is meant by inorder predecessor of a Node in Binary Tree? For which node, Inorder Predecessor is NULL?

- Inorder predecessor of a node is the previous node in Inorder traversal of the Binary Tree.
- Inorder predecessor is NULL for the first node in In order traversal.

94) Difference between Linked List and Queue.

The difference between queues and linked lists is that insertions and deletions may occur anywhere in the linked list, but in queues insertions can be made only in the rear end and deletions can be made only in the front end.

95) Explain the advantage of Doubly Linked List over Singly Linked List

1) A Doubly Linked List can be traversed in both forward and backward direction whereas singly Linked List can be traversed in only one direction.
2) The delete operation in Doubly Linked List is more efficient if the pointer to the node to be deleted is given.
3) We can quickly insert a new node before a given node in Doubly Linked List

96) Define Complete Binary Tree

A complete binary tree is a binary tree in which every level, except possibly the last, is completely filled, and all nodes are as far left as possible.

97) Define the terminologies such as Root of the tree, Leaf of the tree

- Root is the topmost node of the tree.
- Leaf is a node that does not have a child node in the tree.

98) Explain the process of deleting the node that has two childrens in Binary Search Tree?

Find inorder successor of the node. Copy contents of the inorder successor to the node and delete the inorder successor. Note that inorder predecessor can also be used.

99) What is skewed Binary Tree

If a tree which is dominated by left child node or right child node, is said to be a Skewed Binary Tree. In a skewed binary tree, all nodes except one have only one child node. The remaining node has no child.

100) Name the types of Skewed Binary Tree.

There are two types of Skewed Binary Tree. They are

1) Left-Skewed Binary Tree..
2) Right-Skewed Binary Tree.

101) When do we need to split a node in B Tree?

If a node is already full, then a new key insertion will overflow and disturb the B Tree property. It would need re-establishment of the property, which is ideally achieved by splitting the node.

102) What is Left Child Right Sibling Representation of a tree?

It is a different representation of an n-ary tree where instead of holding a reference to each and every child node, a node holds just two references, first a reference to it's first child, and the other to it's immediate next sibling

103) What are all the types of Rotations that can occnr in AVL Tree when the tree become unbalanced?

We might need to perform any of the 4 types of rotations when building AVL Tree in order to make the tree balanced. They are

1) Left Rotation
2) Right Rotation
3) Left-Right Rotation
4) Right-Left Rotation

104) Let the order of B-Tree be 'p'. What is the minimum and maximum number of children that root node (which is also a non-leaf node) can have? What is the minimum and maximum number of keys that root node can have ?

- Root Node should have minimum of **2** children.
- Root Node should have maximum of **p** children.
- Root Node should have minimum of **1** key.
- Root Node should have maximum of **p-1** Keys.

105) Let the order of B-Tree be 'p'. What is the minimum and maximum number of children that Internal node can have?. What is the minimum and maximum number of keys that Internal node can have?

- Internal Node should have minimum of **ceil(p/2)** children.
- Internal Node should have maximum of **p** children.
- Internal Node should have minimum of **ceil(p/2)-1** key.

- Internal Node should have maximum of **p-1** keys.

106) Let the order of B-Tree be 'p'. What is the minimum and maximum number of keys that leaf node can have?

- Leaf Node should have minimum of **ceil(p/2)-1** key.
- Leaf Node should have maximum of **p-1** keys.

107) On What nodes, record pointers are present in B-Tree and in B+Tree

- In B-Tree, Record pointers will be present at leaf nodes as well as on internal nodes
- In B+ tree we will have record pointers only at leaf level.

108) Does all the leaf nodes of B+ tree are connected. If yes, then mention the data structures using which which the are connected.

Yes. All the leaves of B+ Tree are connected. They are connected using Linked List Data Structure.

109) Define Queue

The queue has two ends, the front end and the rear end. The rear end is where we insert elements and front end is where we delete elements. We can traverse in a linear queue in only one direction ie) from front to rear. Queues are also referred as First-In-First-Out (FIFO) Lists.

110) Define Priority Queue

Priority queue is a collection of elements, each containing a key referred as the priority for that element. Elements can be inserted in any order (i.e., of alternating priority), but are arranged in order of their priority value in the queue. The elements are deleted from the queue in the order of their priority (i.e., the elements with the highest priority is deleted first). The elements with the same priority are given equal importance and processed accordingly.

111) Define Double Ended Queue

Double Ended Queue is the another form of queue in which insertions and deletions are made at both the front and rear ends of the queue.

112) Define Circular Queue

Circular Queue is a linear data structure in which the operations are performed based on FIFO (First In First Out) principle and the last position is connected back to the first position to make a circle.

113) How many queues are needed to implement a stack

Two queues are needed to implement the stack.

114) How many queues are needed to implement a priority queue

Two queues are needed to implement a priority queue. One queue is used for actual storing of data and another for storing priorities.

115) What are all the operations in the queue. Explain each of those operations.

In the queue, only two operations are allowed. They are enqueue and dequeue.

- Enqueue operation means to insert an item into the queue,
- Dequeue operation means removing the item from the queue.

116) Define Graph

A graph G consist of a nonempty set V which is a set of vertices of the graph, a set E which is the set of edges of the graph, and a mapping from the set for edge E to a set of pairs of elements of V. It can also be represented as G=(V, E).

117) Define Adjacent Node and Adjacent Edge in a graph.

- Any two nodes which are connected by an edge in a graph are called adjacent nodes.
- Two edges are called adjacent if they have an end vertex in common

118) What is a simple graph?

A simple graph is a graph, which has not more than one edge between a pair of nodes than such a graph is called a simple graph.

119) Name the different ways of representing a graph?

You can represent the graph in two methods. They are

1)Adjacency matrix representation

2) Adjacency list representation.

120) What is a minimum spanning tree?

A minimum spanning tree of an undirected graph G is a tree formed from graph edges that connects all the vertices of G at the lowest total cost.

121) Name two algorithms to find minimum spanning tree?

Two algorithms to find Minimum Spanning Tree is

- Kruskal'salgorithm
- Prim's algorithm

122) What is a Complete Graph

Complete graph is a simple undirected graph in which every pair of distinct vertices is connected by a unique edge.

123) If there are 'n' vertices in Complete Graph G, then how many edges will be present?.

Since every pair of 'n' vertices are connected by unique edge in complete graph, totally n(n-1)/2 edges will be present where n is Number of Vertices.

124) What is meant by strongly connected in a graph?

An undirected graph is connected, if there is a path from every vertex to every other vertex. A directed graph with this property is called strongly connected.

125) What is a Biconnected Graph

Biconnected graph is a connected and nonseparable graph, meaning that if any one vertex were to be removed, the graph will remain connected.

126) Differentiate BFS and DFS.

BFS	DFS
BFS, stands for Breadth First Search.	DFS, stands for Depth First Search.
BFS uses Queue to find the shortest path.	DFS uses Stack to find the shortest path.
BFS is better when target is closer to Source.	DFS is better when target is far from source.
As BFS considers all neighbour so it is not suitable for decision tree used in puzzle games.	DFS is more suitable for decision tree. As with one decision, we need to traverse further to augment the decision. If we reach the conclusion, we won.
BFS is slower than DFS.	DFS is faster than BFS.

127) What is the purpose for which Dijikstra's algorithm is used.

Dijikstra's Algorithm is used for finding Single Source Shortest Path. Single source shortest algorithm is nothing but one node in a graph will taken as source and we will find shortest distance to all the other nodes from source node.

128) What is the purpose for which Floydd Warshall Algorithm is used.

The Floyd-Warshall algorithm is a all source shortest path algorithm for graphs. This means Floydd warshall computes shortest-path between every pair of verices whereas Dijikstra and Bellman-Ford computes the shortest path from a single source.

129) What is Articulation Point in graph.

A vertex in an un-directed connected graph is an articulation point if and only if removing it disconnects the graph. Articulation points represent vulnerabilities in a connected network – single points whose failure would split the network into 2 or more components.

130) Why Prim's Algorithm fails for Directed Graph?

Prim's algorithm assumes that all vertices are connected. But in a directed graph, every node is not reachable from every other node. So, Prim's algorithm fails due to this reason.

131) How do you write Path Matrix for a Graph with 'n' vertices?

Path Matrix is a matrix sized n*n, where n is the number of vertices of the graph. The element on the ith row and jth column is 1 if there's a path from ith vertex to jth vertex in the graph, and 0 if there is not.

132) Define Hashing?

Hashing is an important Data Structure which is designed to use a special function called the Hash function which is used to map a given value with a particular key for faster access of elements. The efficiency of mapping depends of the efficiency of the hash function used.

133) What do you mean by hash table?

The hash table data structure is merely an array of some fixed size, containing the keys. A key is a string with an associated value. Each key is mapped into some number in the range 0 to tablesize-1 and placed in the appropriate cell.

134) What do you mean by collision in hashing?

A situation when the resultant hashes for two or more data elements in the data set U, maps to the same location in the has table, is called a hash collision.

135) What is open addressing?

Open addressing is also called closed hashing, which is an alternative to resolve the collisions with linked lists. In this hashing system, if a collision occurs, alternative cells are tired until an empty cell is found. With this method a hash collision is resolved by probing.

136) What do you mean by separate chaining?

Separate chaining is a collision resolution technique to keep the list of all elements that hash to the same value. This is called separate chaining because each hash table element is a separate chain (linked list). Each linked list contains all the elements whose keys hash to the same index.

137) What do you mean by linear probing in Hashing?

Linear probing is an open addressing collision resolution strategy in which F is a linear function of i, F(i)=i. This amounts to trying sequentially in search of an empty cell. If the table is big enough, a free cell can always be found, but the time to do so can get quite large.

138) What do you mean by quadratic probing in Hashing?

Quadratic probing is an open addressing collision resolution strategy in which F(i)=i^2 (i square). There is no guarantee of finding an empty cell once the table gets half full if the table size is not prime. This is because at most half of the table can be used as alternative locations to resolve collisions

139) What do you mean by primary clustering in Hashing?

In linear probing collision resolution strategy, even if the table is relatively empty, blocks of occupied cells start forming. This effect is known as primary clustering means that any key hashes into the cluster will require several attempts to resolve the collision and then it will add to the cluster.

140) What is meant by Load Factor of Hash Table?

The load factor is the number of keys stored in the hash table divided by the capacity of the hash table (number of slots in the hash table).The size should be chosen so that the load factor is less than 1.

141) What is the worst case time complexity of "Search operation" in Binary Search Tree and AVL Tree where both Binary Search Tree and AVL Tree contains 'n' elements?

Search operation:-

- Binary Search Tree – O(n)
- AVL Tree – O(log n)

142) Write the disadvantages of separate chaining.

• The elements are evenly distributed. Some elements may have more elements and some may not have anything.

• It requires pointers. This leads to slow the algorithm down a bit because of the time required to allocate new cells, and also essentially requires the implementation of a second data structure.

143) What do you mean by rehashing?

If the table gets too full, the running time for the operations will start taking too long and inserts might fail for open addressing with quadratic resolution. A solution to this is to build another table that is about twice as big with the associated new hash function and scan down the entire original hash table, computing the new hash value for each element and inserting it in the new table. This entire operation is called rehashing.

144) What is heap in algorithm?

Heap is a special case of balanced binary tree data structure where the root-node key is compared with its children and arranged accordingly.

145) What is the property min-Heap?

The value of each node is greater than or equal to the value of its parent, with the minimum value at the root.

146) What is the property of max-Heap?

The value of each node is less than or equal to the value of its parent, with the maximum value at the root.

147) Is Heap a Full Binary Tree ? If no, Explain Why.

Heap is not a Full Binary Tree. Its is a Complete Binary Tree. Because While building Heap, all the elements are stored from Left to Right and it resembles the property of Complete Binary Tree.

148) Explain Internal Sorting

An internal sort is any data sorting process that takes place entirely within the main memory of a computer. This is possible whenever the data to be sorted is small enough to all be held in the main memory.

149) Explain External Sorting

External sorting is a term for a class of sorting algorithms that can handle massive amounts of data. External sorting is required when the data being sorted do not fit into the main memory of a computing device (usually RAM) and instead they must reside in the slower external memory (usually a hard drive)

150) Define bubble sort

Bubble sort is a simple sorting algorithm that works by repeatedly stepping through the list to be sorted, comparing each pair of adjacent items and swapping them if they are in the wrong order. The pass through the list is repeated until no swaps are needed, which indicates that the list is sorted. The algorithm gets its name from the way smaller elements "bubble" to the top of the list.