

JYOTHY INSTITUTE OF TECHNOLOGY

BENGALURU



Department of Information Science and Engineering

Data structures and applications(18CS33)

General questions and Module-1

1. What is a computer? List out few real time applications of computer.
2. List out the categories of languages. Explain
3. What language a computer can understand?
4. What is the difference between a Computer and a normal electronic device?
5. What is a variable?
6. What is a data type?
7. What is a pointer? Explain how to declare initialize and access a pointer variable with example.
8. Questions on predicting the output of a given program.
9. What are the advantages of pointers?
10. What is dynamic memory allocation? List out the functions which will help is in allocating memory dynamically with example.
11. Programs on dynamic memory allocation.
12. What is a string? Explain how to declare initialize and access a string.
13. List out five strings built in functions.
14. Programs on String manipulation functions. Both using normal array and pointers.
15. How to access the array both 1d and 2d using pointers (includes programs as well).
16. Discuss about Call by value and Call by reference.
17. What is a data structure? Discuss the categories of data structure.
18. Define a structure.
19. Discuss the difference between structure and an array.
20. Discuss the difference between structure and a union.
21. Discuss about the operations to be performed on array with programs.
22. What is a stack?
23. Discuss about the applications of stack.
24. Give the implementation of stack using functions.
25. Give the algorithm to covert the given infix expression to a postfix expression with an example.

All Possible Questions for Stacks,Queues,Linked Lists,Trees and Graphs

a.	Define Stack.	06
b.	Write a program for stack of integers.	08
a.	Define Queue? Give its ADT.	06
b.	Write a program for Queue of integers.	08
a.	Write an algorithm to convert infix to postfix expression and solve : $a*b+c-d/f$.	10
b.	Give the stack ADT including observer functions.	04
a.	Write an algorithm and program to perform parenthesis matching.	10
b.	Give the Queue ADT including observer functions.	04
a.	Explain the Queue data structure and list few of its applications.	06
b.	Write a program for Queue of integers.	08
a.	Explain the stack data structure and list few of its applications.	06

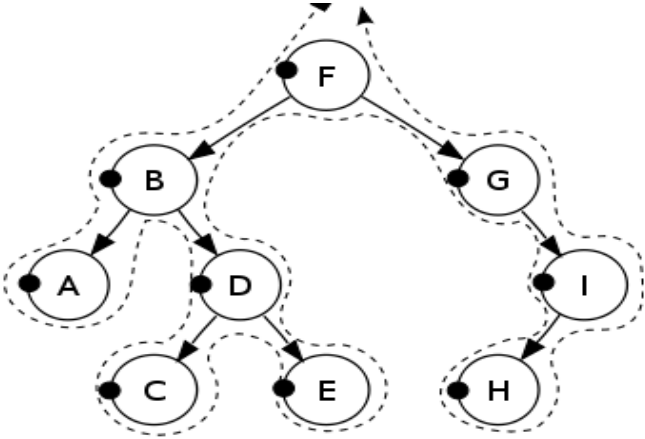
b.	Write a program for stack of characters to reverse a string.	08
a.	Compare Stack and Queue data structure.	06
b.	Write functions for Push and Pop operations of a stack.	08
a .	Write an algorithm and program to perform parenthesis matching.	10
b.	Write functions for Enqueue and Dequeue.	04
a.	Write functions for Enqueue and Dequeue.	07
b.	Write functions for Push and Pop operations of a stack.	07
a.	Compare Stack and Queue data structure.	06
b.	Write a program for the implementation of Queue of integers.	08
a.	Write an algorithm to convert infix to postfix expression and solve : A * (B + C * D) + E by showing the contents of stack.	08
b.	List out the applications of stacks and queues.	06

a.	What is a linked list? Give the structure of linked list which can hold student record.	08
b.	Explain the operation of delete at specified position using neat diagram.	06
	Write a C Program to perform following operations on Singly Linked List ADT : <ul style="list-style-type: none"> i. Create ii. Insert iii. Delete iv. Display 	14

	<p>Write a C Program to perform following operations on Doubly Linked List ADT :</p> <ul style="list-style-type: none"> i. Create ii. Insert iii. Delete iv. Display 	14
5	<p>Write a C Program to perform following operations on Circular Linked List ADT :</p> <ul style="list-style-type: none"> i. Create ii. Insert iii. Delete iv. Display 	14
a.	Write a C program to perform insert at specified position for a singly linked list. Explain the same with a neat diagram.	07
b.	Write a C program to perform delete at specified position for a singly linked list. Explain the same with neat diagram.	07
a.	Write a C program to perform insert at specified position for a doubly linked list. Explain the same with neat diagram.	07
b.	Write a C program to perform delete at specified position for a doubly linked list. Explain the same with neat diagram.	07
a.	Give the linked list ADT. List few of the applications of linked list.	06
b.	Write a function to merge two linked list.	08
a.	Write a function to find the mid element of a linked list.	07
b.	Write a recursive function to display the list elements.	07
a.	<p>Write a function for Singly linked list.</p> <ul style="list-style-type: none"> i. Insert front ii. Insert Rear 	07

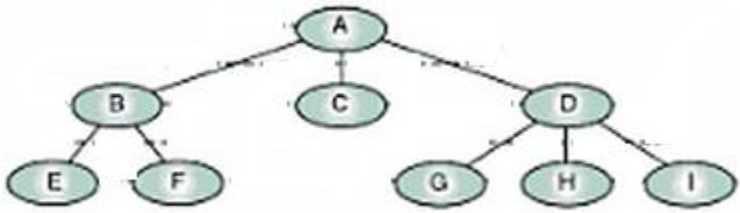
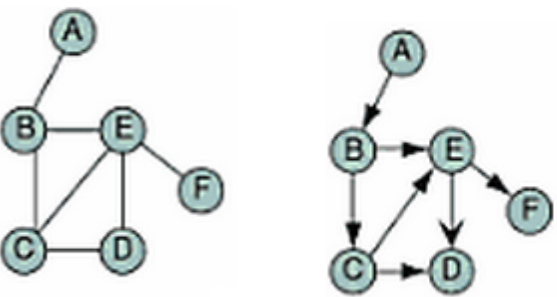
b.	Write a functionsfor Doubly linked list .	
	i. Delete Front ii. Delete Rear	07
a.	Write a program to find the nth element from the last in a given singly linked list.	07
b.	Write a program to split the given list into two lists.	07
a.	Write a function to find the mid element of a linked list.	07
b.	Give the linked list ADT. List few of the applications of linked list.	07
	Define a general linear list, the basic operations on lists, their applications and show its implementation.	14
	Assume that a college has a general linear list that holds information about the students and that each data element is a record with three fields: ID, Name and Grade. Write an Algorithm or Program that helps a professor to change the grade for a student. The delete operation removes an element from the list, but makes it available to the program to allow the grade to be changed. The insert operation inserts the changed element back into the list. The element holds the whole record for the student, and the target is the ID used to search the list.	14

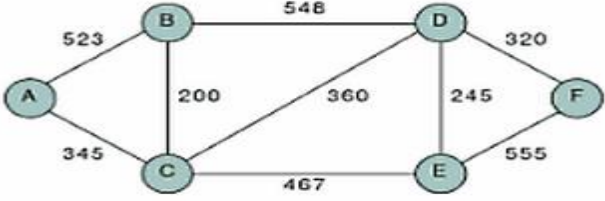
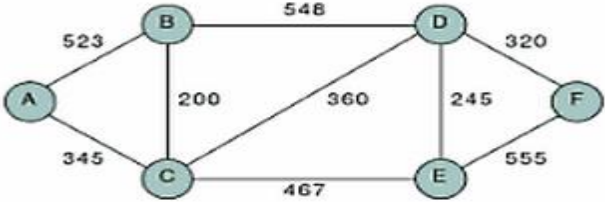
	Discuss stack data structure and implement stack using linked list? Justify the advantage of using list to implement the stack.	14
	Discuss Queue data structure and implement Queue using linked list? Justify the advantage of using list to implement the stack.	14
	Discuss Non linear data structure and implement Binary Search Tree, with Pre-order, In-order and Post-order traversal functionalities.	14
a.	What is a tree? Write an algorithm or program for tree traversals.	08
b.	Write an algorithm or program for depth first traversal of a tree.	06
	What is binary search tree? Write program to create a binary search tree with the functions to find the largest element in a tree and smallest element in a tree.	14

a.	Write an algorithm or program to search for a given element in the BST.	07
b.	Write an algorithm or program to insert a given element to the BST.	07
a.	Define Binary Search Tree? List few of its applications.	06
b.	Write an algorithm or program to delete a given element from the BST.	08
a.	Write an algorithm or program to perform breadth first traversal.	08
b.	List the advantages and disadvantages of implementing a stack or queue data structure as using linked list.	06
a.	Write an algorithm or program to search for a given element in the BST.	08
b.	<p>Perform the In-order Pre-order and post-order traversal for the given tree.</p> 	06
a.	Define Binary Search Tree? List few of its applications.	06
b.	Write an algorithm or program to search for a given element in the BST.	08
	Implement a Stack using List and using your stack write an application program which helps in converting a given decimal number into binary number.	14
	Define a stack, the basic operations on stacks, their applications and how they can be implemented.	14

	Define a queue, the basic operations on queue, their applications and how they can be implemented.	14
--	---	-----------

a.	What is heap? Define min-heap and max-heap.	06
b.	Explain with an example how to implement heap using arrays.	08
a.	Write an algorithm for reheap-Up operation and explain.	07
b.	Write an algorithm for reheap-Down operation and explain.	07
a.	Write an algorithm for build Heap operation and explain.	07
b.	Give a structure for heap data structure.	07
a.	Write an algorithm for deleteHeap operation and explain.	07
b.	What is heap? Define min-heap and max-heap.	07
	Give the Heap ADT and implement the same in C.	14
	What is a priority queue? Implement Priority Queue in C.	14
a.	Write an algorithm for deleteHeap operation and explain.	07
b.	Write an algorithm for reheap-Down operation and explain.	07
a.	Explain with an example how to implement heap using arrays.	08
b.	Give a structure for heap data structure.	06
a.	Write an algorithm for re heap-Up operation and explain.	07
b.	Write an algorithm for build-Heap operation and explain.	07

a.	What is graph? What are the different kinds of graph explain briefly.	08
b.	With neat diagram show how the following operations on graph are done. i. Insert vertex ii. Delete vertex iii. Find vertex	06
	Explain how graph traversals are done. For the graph given below explain how Depth first and Breadth first traversal is done.  <pre> graph TD A((A)) --- B((B)) A --- C((C)) A --- D((D)) B --- E((E)) B --- F((F)) D --- G((G)) D --- H((H)) D --- I((I)) </pre>	14
	Write a note on Graph storage structure and write the adjacency matrix for the given graph below: a. Undirected Graph b. Directed Graph  <pre> graph LR subgraph "a. Undirected Graph" A --- B B --- C B --- E C --- D C --- E D --- E E --- F end subgraph "b. Directed Graph" A --> B B --> C B --> E C --> D C --> E D --> E E --> F end </pre>	14

a.	<p>Explain what do you mean by minimum spanning tree? Why is it required?</p>	06
b.	<p>Find the minimum spanning tree for the graph given below show each step clearly.</p> 	08
a.	<p>Using Dijkstras shortest path algorithm find the shortest path from A to all other vertex for the given graph below, show each step of clearly.</p>  <p>Explain the how graph can be represented as adjacency list with example</p>	08
b.	<p>With neat diagram show how the following operations on graph are done.</p> <ol style="list-style-type: none"> Insert vertex Delete vertex Find vertex <p>Explain the importance of minimum spanning tree with example.</p>	06

b.		08
	Write short notes on: i. Adjacency Matrix ii. Adjacency list iii. Breadth first traversal iv. Depth first traversal	04 04 03 03
	Write short notes on: i. Spanning tree ii. Graph storage structure iii. Graph Traversals	04 06 04

**All the best! From takeiteasy_engineers and
JIT,Bengaluru**

