MUTHAYAMMAL ENGINEERING COLLEGE

(An Autonomous Institution)

(Approved by AICTE, New Delhi, Accredited by NAAC & Affiliated to Anna University)
Rasipuram - 637 408, Namakkal Dist., Tamil Nadu

Estd. 2000

IQAC

MUST KNOW CONCEPTS

MKC

IT

2020-2021

SUBJECT		19CSC01-Data Structures and Algorithms		
S.NO	**TERM**	**Notation (Symbol)**	**Concept/Definition/Meaning/Units/Equation/Expression**	**Units**
UNIT – I - INTRODUCTION				
1	Data		Data are simply values or sets of values	
2	Information		Processed Data	
3	Datum		Singular form of Data	
4	Data		Plural form of Data	
5	Data structures	DS	Way of organizing data in a computer called DS	
6	Classification of DS		Static data structures Dynamic data structures	
7	Static data structures		Fixed size data structure.EX: Array, pointers, structures	
8	Dynamic data structures		Variable size data structure. Ex: linked lists, stacks, queues, trees	
9	Types of data structure		Linear data structure. Non-linear data structure	
10	Linear data structures		Data are arranged in sequential order	
11	Non-linear data structure		Data structures that don't have a linear relationship between its adjacent elements but have a hierarchical relationship	
12	Abstract data type	Adt	Set of operations for which the implementation of the data structure is not specified	
13	Primitive data types		Each variable has a specific data typeit tells - size, range called primitive data types	
14	4 basic primitive data types		Integer , floating-point ,character and Pointer	
15	Pointer		Special type of variables that are used to store	

			address of another variable	
16	Searching		Finding an element position in a given array called searching type: linear search binary search	
17	Efficiency of DS		Efficient Algorithm that takes least possible running time and consumes least memory space	
18	Asymptotic analysis		Measures the performance of the algorithm with the change in the order of the input size	
19	Case complexity		Worst case complexity, best case complexity and average case complexity	
20	Asymptotic complexity		Approximate measure of time complexity is called Asymptotic complexity	
21	Asymptotic notations		Is measured with the help of asymptotic notations	
22	Time complexity		Quantifies the amount of time taken by an algorithm to run as a function	
23	List of Asymptotic Notations		Theta notation, Omega notation and Big-O notation	
24	Logn		A big problem is solved by cutting the original problem in smaller sizes, by a constant fraction at each step	
25	N (linear)		A small amount of processing is done on each input element	
UNIT II - STACKS AND QUEUES				
26	Array		Fixed-size DS	
27	Recursion function		Recursion is an approach in which a function calls itself with an argument	
28	Stack		Stack is an ordered collection of elements in which insertions and deletions are restricted to one end called top	
29	Top		Insertions and deletions of stack take place in top pointer	
30	Push operation		Inserting an element in stack	
31	Pop operation		Removing an element from stack	
32	Peek operation		Viewing top element of stack	
33	Empty stack		If top=-1 represent empty stack	

34	Ful		If top=maxsize-1 represent full stack	
35	Queue		Queue is an ordered collection of elements in which insertions and deletions take place in 2 ends	
36	Rear end		The end from which elements are added referred to rear end	
37	Front end		End from which deletions are made is referred to as the front end	
38	Priority queue		Priority queue is a collection of elements, each containing a key referred as the priority for that element	
39	Enqueue		Inserting an element in queue	
40	Dequeue		Removing an element from queue	
41	Front		Ptr points to 1,st element of queue	
42	Rear		Ptr points to last element of queue	
43	Types of queues		Linear queues Circular queues Priority queue	
44	Applications of stacks		Reversing a string Balanced parenthesis Evaluation of arithmetic expressions	
45	Underflow		Checking queue is empty (contain no elements in array) called underflow	
46	Overflow		Checking queue is full (contain all elements in array) called overflow	
47	LIFO		Last in first out (principle followed by stack)	
48	FIFO		First in first out(principle followed by stack queue)	
49	Max heap		The key at root must be maximum among all keys present in binary heap	
50	Min heap		The key at root must be minimum among all keys present in binary heap	
UNIT III - LINKED LIST				
51	Structure		Structure is a collection of variables belongings to the different data type	
52	Dynamic memory allocation		The process of allocating memory at runtime is known as dynamic memory allocation	
53	Malloc()		Allocates requested size of bytes in memeory	
54	Free		Releases previously allocated memory	
56	Realloc		Modify the size of previously allocated space	

57	Singly linked list		Linked list elements are not stored at contiguous location	
58	Doubly linked list		Contains an extra pointer, typically called previous pointer, together with next pointer and data	
59	Circularly linked list		Linked list where all nodes are connected to form a circle. There is no null at the end	
60	Operations of linked list		Creation, insertion(in first, middle and last), deletion(in first, middle and last),searching, traversing	
61	Application of linked list		Polynomial manipulation Stacks Queues	
62	Infix notation		X + Y ,Operators are written in-between their operands	
63	Postfix notation		X Y +, Operators are written after their operands.	
64	Prefix notation		+ X Y, Operators are written before their operands	
65	Other name for Postfix notation		Reverse Polish notation	
66	Other name for Prefix notation		also known as "Polish notation	
67	Post fix expression for (a+b*c)/d		abc*+d/	
68	Pre fix expression for (a+b*c)/d		/+a*bcd	
69	Head		First node of list	
70	Fields of Single linked list node		Data and next	
71	Next		Address of next node of list	
72	Fields of Double linked list node		Data, next and previous	
73	previous		Address of previous node of list	
74	Isempty of list ()		If head== NULL represent empty list	
75	Traversing		Operation perform viewing of all element in the list	
UNIT IV-TREES				
76	Tree		A tree is a non-linear data structure, which represents hierarchical relationship between individual data items	
77	Height of a Tree		Length of the longest path from the root to a leaf	
78	Path in a tree		Sequence of distinct nodes in which successive nodes are connected by edges	
79	Leaf node		A node that has no children	

80	Binary tree nodes		A binary tree is a tree in which every non-leaf node has atmost two children	
81	Full binary tree		A full binary tree is a tree in which all leaves are on the same leve	
82	Complete binary tree		Is a binary tree in which every level, except possibly the last, is completely filled	
83	Right-skewed binary tree		Binary tree is a tree, which has only right child nodes	
84	**Representing a binary tree**		Linear representation using arrays. Linked representation using pointers.	
85	**Tree traversal**		Moving through all the nodes in the binary tree	
86	Types of tree traversal		Preorder traversal Inorder traversal Postorder traversal	
87	Tasks performed for traversing a binary tree		Visiting a node. Traverse the left subtree Traverse the right subtree	
88	Preorder traversal		Process the root node Traverse the left subtree Traverse the right subtree.	
89	Inorder traversal		Traverse the left subtree. Process the root node. Traverse the right subtree	
90	Postorder traversal		Traverse the left subtree Traverse the right subtree. Process the root node	
91	Binary search tree		Binary tree, in which , the values in any left subtree is less than the value of its parent node, the values in any right subtree is greater than the value of its parent node and the left and right subtrees of each node are again binary search trees	
92	Property of heap		Structure property Heap property	
93	Structure property		It is a complete binary tree.	
94	Heap property		Heap property - For a "max heap", the property is that the value of each node is always less than or equal to the value of its parent.	
95	Root		In a tree data structure, the first node is called as Root Node	
96	Parent node		The node which has child / children	

97	Siblings		nodes which belong to same Parent	
98	Degree		total number of children of a node is called as DEGREE of that Node	
99	AVL Tree		Balanced Binary search tree	
100	Balanced factor		Height of left subtree- Height of right subtree	
UNIT V- SORTING AND HASHING				
101	Hashing		Searching technique in O(1) time complexity	
102	Hash function		Hash_key=key mod tablesize	
103	Collision in hashing		When an element is inserted, it hashes to the same value as an already inserted element, and then it produces collision.	
104	Separate chaining		Separate chaining is a collision resolution technique to keep the list of all elements that hash to the same value	
105	Open addressing		Open addressing is a collision resolving strategy in which, if collision occurs alternative cells are tried until an empty cell is found	
106	Types of collision resolution strategies in open addressing		Linear probing Quadratic probing	
107	Probing		Process of getting next available hash table array cell	
108	Linear probing		F(i)=i. Hi(x)=(hash(x)+f(i))mod tablesize . I=1,2,3,4…..	
109	Quadratic probing		F(i)=i^2. Hi(x)=(hash(x)+f(i))mod tablesize . I=1,2,3,4…..	
110	Sorting		A sorting algorithm is used to rearrange a given array or list elements in ascending or descending order.	
111	Types of internal sorting		Bubble Sort Insertion Sort Selection Sort Quick Sort Merge Sort Heap Sort	
112	Classification of sorting		Internal sorting and external sorting	
113	Internal sorting		internal sorting the data that has to be sorted will be in the main memory	
114	External sorting		External sorting it will on disks, outside main	

			memory	
115	Types of external sorting		Two-way merge sort ,radix sort	
116	Time complexity of bubble sort		Θ (n)	
117	Divide-and-conque		Divide: Break the given problem into subproblems of same type. Conquer: Recursively solve these subproblems Combine: Appropriately combine the answers	
118	Not a stable sorting algorithm		Bubble sort	
119	Not a stable sorting algorithm		Merge sort	
120	O(nlogn)		Running merge sort on an array of size n which is already sorted is	
121	O(n log n))		The time complexity of a quick sort algorithm	
122	Time complexity of insertion sort		Θ (n)	
123	Mod function %		Returns remainder value	
124	7%8		7	
125	10%8		2	
			Technical Questions	
126	Last in last out		Stack is also called as	
127	Queue		……. Is a pile in which items are added at one end and removed from the other	
128	Stack		…… is very useful in situation when data have to stored and then retrieved in reverse order	
129	Stack		DS used for depth first traversal	
130	Queue		What data structure is used in breadth first search of a graph to hold nodes	
131	Dequeues		A …….. is a linear list in which insertions and deletions are made to from either end of the structure.	
132	ABDECF		The post-order traversal of the binary tree is DEBFCA. Find out the pre-order traversal	
133	Algorithm used to find minimum spanning tree		ruskal's algorithmPrim's algorithm	
134	Dijkstra algorithm		Algorithm used to find shortest path in graph	
135	floyd-warshall all pairs shortest path algorithm		algorithm computes the shortest paths between each pair of nodes	
136	single source		Dijkstra algorithm is also called the ……. shortest path problem	

137	binary search trees		The in-order traversal of the tree will yield a sorted listing of elements of tree in ……..	
138	Edge begins at u and ends at v		In a graph if e=(u,v) means	
139	Overflow		Before inserting into stack one must check the condition	
140	double ended queue		The another name of dequeue is ……	
141	Underflow		efore deletion condition into stack …… has to be checked.	
142	Front=Null		The condition …….. indicate the queue is empty	
143	Front=Rear		The condition …….. indicate the queue has one node is	
144	top		The pointer associated with the stack is ……..	
145	Selection		If the number of records to be sorted is small, then …… sorting can be efficient.	
146	running time		The complexity of the sorting algorithm measures the …… as a function of the number n of items to be sorter	
147	Selection sort		Which of the following sorting algorithm is of priority queue sorting type	
148	quick sort		Partition and exchange sort is ……..	
149	Merge sort		Which of the following sorting algorithm is of divide and conquer type?	
150	Dircted Acyclic Graph		connected graph T without any cycles is called ……..	

Faculty Team Prepared	**Dr.E.Punaselvam**	**Signatures**

HoD