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.

MUST KNOW CONCEPTS

MKC

DEPT IT

2020-21

Course Code & Course Name : 19ITC11 & Design and Analysis of Algorithm

Year/Sem/Sec : II/IV/-

S.No.	Term	Notation (Symbol)	Concept / Definition / Meaning / Units / Equation / Expression	Units
			Unit-I : Introduction	
1.	Algorithm		Sequence of instructions for solving a problem	
2.	pseudo code		Mixture of a natural language and programming language	
3.	Time efficiency		How much amount of time needed to execute	
4.	Space efficiency		How much amount of space needed to execute	
5.	Exact Algorithm		Solving the problem exactly	
6.	Approximate Algorithm		solving it approximately	
7.	sorting problem		Rearrange the items of a given list in non decreasing order	
8.	searching problem		Finding a given value,	
9.	Analysis Framework		1.Measuring an Input's Size 2. Units for Measuring Running Time 3. Orders of Growth 4. Worst-Case, Best-Case, and Average-Case Efficiencies 5. Recapitulation of the Analysis Framework	
10.	O-notation		$t(n) \le cg(n)$ for all $n \ge n0$.	
11.	Ω-notation		$t(n) \ge cg(n)$ for all $n \ge n_0$.	
12.	ɵ -notation		$c_2 g(n) \le t(n) \le c_1 g(n)$ for all $n \ge n_0$.	
13.	Asymptotic Notations		• O-notation • Omagha -notation • ɵ -notation	

14.	Fundamental Data Structures		• Linear Data Structures • Graphs • Trees	
15.	Vertices		a collection of points	
16.	Edges		A collection of points connected by line segments	
17.	Characteristics of Algorithm		Simplicity, Time consuming, easy to understand, generality.	
18.	Methods specifying for an algorithm		Flow chart, Natural language, Program	
19.	Understanding the Problem		It is the first step in solving the problem	
20.	The main measure for efficiency algorithm are		Time and space	
21.	Algorithmic analysis count		The number of arithmetic and the operations that are required to run the program	
22.	The concept of order Big O is important because		It can be used to decide the best algorithm that solves a given problem	
23.	Non-recursive function		Does not references itself	
24.	Recursive function		Function which calls itself again and again	
25.	What are the case does exist in complexity theory		Best case,Worst case,Average case	
Unit-II : Brute force and Divide-and-Conquer				
26.	Brute force method		• Straight forward approach • Method has “just do it” approach • Useful for solving smaller program	
27.	Applications of bruteforce method		Selection sort, bubble sort, sequential sort, Assignment problem	
28.	Closest pair problem		Find the closest point in set of n points	
29.	convex		A set of points in the plane	
30.	Convex-hull		The convex hull of a set *S* of points is the smallest convex set containing *S*	
31.	Exhaustive search		It requires searching all the possible solution for the best solution	

32.	Exhaustive search- Applications		Travelling Salesman, Knapsack problem, Assignment problem	
33.	Travelling Sales man Problem	TSP	The problem is to find the shortest possible route.	
34.	Hamiltonian circuit		A cycle that passes through all the vertices of the graph exactly once.	
35.	Eight-queens problem		Classic puzzle of placing eight queens on an 8 × 8 chessboard	
36.	Vertices represents in TSP		cities	
37.	Edges represents in TSP		Weight or distance	
38.	Divide and Conquer method		Smaller sub problems, sub problems are solved recursively	
39.	Applications of divide and conquer		Binary search, quick sort, merge sort, multiplication of large integers	
40.	Searching types		Linear, binary	
41.	Linear search		To find a particular value and not in sorted order	
42.	Application of Graphs:		Physics and Chemistry, Mathematics, Social Science	
43.	Mid value in binary search		mid = (low + high) /2, low-0^{th} value and high-last value	
44.	Merge sort		Merge Sort is a sorting algorithm. Merge Sort is a divide and conquer algorithm.	
45.	Quick sort		select an element as pivot, partition the array around pivot and recurse for subarrays on left and right of pivot.	
46.	Strassen algorithm		It is faster than the standard matrix multiplication algorithm	
47.	Assignment problem		Assign a number of jobs to an equal number of machines so as to minimize the total assignment cost for execution of all the jobs	
48.	Binary search working		Binary search works by dividing the array into 2 halves around the middle element	

49.	Graph		Consists of a set of vertices, and set of edges	
50.	Graph types		BFS,DFS	
Unit-III : Dynamic Programming and Greedy Technique				
51.	Dynamic programming		Reduce the time complexity, provide optimal solution	
52.	Advantages of dynamic programming		Computing Fibonacci numbers, completing binomial coefficient	
53.	Applications of dynamic programming		Find shortest path between all pair of vertices	
54.	Warshalls algorithm		Solve all pair shortest path problem	
55.	Floyds algorithm		Find optimal solution	
56.	Greedy technique used in		Minimum spanning tree, shortest path problem	
57.	Applications for greedy technique		Huffman coding is a lossless data compression algorithm.	
58.	Huffman Algorithm		which assigns codewords of different lengths to different symbols,	
59.	Variable-length encoding		8 bits	
60.	Huffman code		A Huffman code is an optimal prefix tree variable-length encoding technique which assign bit strings to characters based on their frequency in a given text.	
61.	Minimum spanning tree		Divide and conquer	
62.	Which strategy merge sort using		O(n2)	
63.	Complexity of merge sort algorithm		Pivot element	
64.	The running time of quick sort depends heavily on the selection of		O(n2)	
65.	The worst-case time complexity of Quick Sort		O(n logn)	
66.	The worst-case time complexity of Merge		Bubble sort	

	Sort			
67.	Which of the sorting procedures is the slowest		Counting the maximum memory needed by the algorithm	
68.	The space factor when determining the efficiency of algorithm is measured by		Insertion sort	
69.	The way a card game player arranges his cards as he picks them one by one can be compared to		Solve a problem by using top down approach	
70.	Memory function		provides the smallest possible search time	
71.	optimal binary search tree	OBST	bottom-up, and solving all the sub-problems only once.	
72.	Memory function		has the minimum sum of weights among all the trees that can be formed from the graph.	
73.	Prim's algorithm		Prim's algorithm is a greedy and efficient technique, which is used to find the minimum spanning the tree of a weighted linked graph.	
74.	Time complexity of the Huffman algorithm		Reduce the time complexity, provide optimal solution	
75.	Memory function		has the minimum sum of weights among all the trees that can be formed from the graph.	
Unit-IV : Iterative Improvement and Limitation of algorithm				
76.	Iterative improvement		This techniques build an optimal solution by iterative refinement	
77.	Linear programming		To optimize linear function of several variables	
78.	Bipartite Graph		No two edges share an end point	
79.	maximum matching		maximum matching is a matching of maximum size (maximum number of edges)	
80.	Stable marriage problem		Identifying stable matching between two sets of elements	

81.	Simplex method		It is an approach to solving linear programming models	
82.	Decision tree		It is a tree-like graph or model of decisions	
83.	Decision tree uses		For searching and sorting	
84.	Optimization problem		To maximize or minimize some values.Ex: Finding the shortest path between two vertices in a graph.	
85.	Polynomial time algorithm.		For input size ***n***, if worst-case time complexity of an algorithm is $O(n^k)$, where ***k*** is a constant	
86.	NP Hard problems		• The circuit-satisfiability problem • Set Cover • Vertex Cover • Travelling Salesman Problem	
87.	NP complete problem		No polynomial time algorithm	
88.	P-class		Problems are solvable in polynomial time	
89.	NP-class		Problems are verifiable in polynomial time.	
90.	Lower Bound Theory Base Bound Theory		Calculation of minimum time that is required to execute an algorithm	
91.	Techniques in lower bound theory		• Comparisons Trees. • Oracle and adversary argument • State Space Method	
92.	Graph coloring problem		Neighbour node don't have same color	
93.	Backtracking problem		To solve combinational problem, optimization problem, decision problem	
94.	Maximum Flow problem		Maximum amount of flow that the network would allow to flow from source to sink.	
95.	Basic solution for simplex method		At most m non zero values for the variables	
96.	A matching in a Bipartite Graph		no two edges share an endpoint.	
97.	Limitation of algorithm		Time consuming, big tasks are difficult to put in algorithm	
98.	Iterative improvement follows which technique		Greedy technique	

99.	Iterative improvement mainly used for		Smaller problems	
100.	Base Bound Theory		Calculation of minimum time for execute a algorithm	
Unit-V : Backtracking ,Branch and Bound and Approximation Algorithm				
101.	**Backtracking**		Depth-first node generation with bounding method.	
102.	Which method used to find Hamiltonian circuit		Backtracking	
103.	N - Queens problem		The problem is to area n-queens on an n-by-n chessboard so that no two queens charge each other by being same row or in the same column or the same diagonal.	
104.	Subset Sum Problem		sum of the elements of subset's' is equal to some positive integer 'X.'	
105.	Assignment problem		Assign a number of jobs to an equal number of machines so as to minimize the total assignment cost for execution of all the jobs	
106.	Travelling Sales man Problem	TSP	The problem is to find the shortest possible route.	
107.	Branch and bound		which is generally used for solving combinatorial optimization problems.	
108.	Application of assignment problem		It involves assignment of people to projects, jobs to machines, workers to jobs and teachers to classes etc	
109.	The worst-case efficiency of solving a problem in polynomial time is		O(p(n))	
110.	Tractable		Problems that can be solved in polynomial time are known as?	
111.	NP		the class of decision problems that can be solved by non-deterministic polynomial algorithms	
112.	Un decidable problems		Problems that cannot be solved by any algorithm	
113.	Example of un		Halting problem	

	decidable problem			
114.	Backtracking problem		To solve combinational problem, optimization problem, decision problem	
115.	Applications of travelling sales man problem		planning, scheduling, logistics and packing	
116.	Approximation problem		Near optimal solution for problem	
117.	Examples for backtracking		Puzzles such as eight queens puzzle, crosswords, verbal arithmetic, Sudoku, and Peg Solitaire.	
118.	Backtracking applications		Electrical engineering, Robotics, Artificial Intelligence, Network communication	
119.	Backtracking technique used in		N Queens problem, sum of subset, Sudoku puzzle, Hamiltonian cycle	
120.	NP hard problem		Algorithm for solving it can be translated into one for solving any NP-problem (nondeterministic polynomial time)	
121.	2-approximation algorithm		Returns a solution whose cost is at most twice the optimal	
122.	Examples of NP problem		integers, rearrange the numbers	
123.	Base Bound Theory		Calculation of minimum time for execute a algorithm	
124.	NP Hard problems		• The circuit-satisfiability problem • Set Cover • Vertex Cover • Travelling Salesman Problem	
125.	NP complete problem		No polynomial time algorithm	
Placement Questions				
126.	Three times the first of three consecutive odd integers is 3 more than twice the third. The third integer is:		Let the three integers be x, $x + 2$ and $x + 4$. Then, $3x = 2(x + 4) + 3 \Leftrightarrow x = 11$. $\therefore$ Third integer $= x + 4 = 15$.	
127.	Look at this series: 7, 10, 8, 11, 9, 12, ...		This is a simple alternating addition and subtraction series. In the first pattern, 3 is added; in the second, 2 is subtracted.	

128.	Look at this series: 22, 21, 23, 22, 24, 23, ….		In this simple alternating subtraction and addition series; 1 is subtracted, then 2 is added, and so on.	
129.	Look at this series: 53, 53, 40, 40, 27, 27, ...		In this series, each number is repeated, then 13 is subtracted to arrive at the next number.	
130.	Look at this series: 1.5, 2.3, 3.1, 3.9, ...		In this simple addition series, each number increases by 0.8.	
131.	Three times the first of three consecutive odd integers is 3 more than twice the third. The third integer is:		Let the three integers be x, $x + 2$ and $x + 4$. Then, $3x = 2(x + 4) + 3 \quad \Leftrightarrow \quad x = 11$. $\therefore$ Third integer = $x + 4 = 15$.	
132.	Look at this series: 7, 10, 8, 11, 9, 12, ...		This is a simple alternating addition and subtraction series. In the first pattern, 3 is added; in the second, 2 is subtracted.	
133.	Look at this series: 22, 21, 23, 22, 24, 23, ...		In this simple alternating subtraction and addition series; 1 is subtracted, then 2 is added, and so on.	
134.	(112 x 5^4) = ?		(112 x 5^4) = 112 x(10)4=112 x 10^4=1120000= 7000022^{4}16	
135.	It was Sunday on Jan 1, 2006. The day of the week Jan 1, 2010 is		On 31st December, 2005 it was Saturday. Number of odd days from the year 2006 to the year 2009 = (1 + 1 + 2 + 1) = 5 days. $\therefore$ On 31st December 2009, it was Thursday. Thus, on 1st Jan, 2010 it is Friday.	
136.	Today is Monday. After 61 days, it will be:		Each day of the week is repeated after 7 days. So, after 63 days, it will be Monday. $\therefore$ After 61 days, it will be Saturday.	
137.	If 6th March, 2005 is Monday,The day of the week on 6th March, 2004 is		The year 2004 is a leap year. So, it has 2 odd days. But, Feb 2004 not included because we are calculating from March 2004 to March 2005. So it has 1 odd day only. $\therefore$ The day on 6th March, 2005 will be 1 day beyond the day on 6th March, 2004.	

			Given that, 6^{th} March, 2005 is Monday. ∴ 6^{th} March, 2004 is Sunday (1 day before to 6^{th} March, 2005).	
138.	The days in x weeks x days?		x weeks x days = $(7x + x)$ days = $8x$ days.	
139.	On 8^{th} Feb, 2005 it was Tuesday. The day of the week on 8^{th} Feb, 2004 is		The year 2004 is a leap year. It has 2 odd days. ∴ The day on 8^{th} Feb, 2004 is 2 days before the day on 8^{th} Feb, 2005. Hence, this day is Sunday.	
140.	The greatest number that will divide 43, 91 and 183 so as to leave the same remainder in each case.		Required number = H.C.F. of (91 - 43), (183 - 91) and (183 - 43) = H.C.F. of 48, 92 and 140 = 4.	
141.	The H.C.F. of two numbers is 23 and the other two factors of their L.C.M. are 13 and 14. The larger of the two numbers is:		Clearly, the numbers are (23 x 13) and (23 x 14). ∴ Larger number = (23 x 14) = 322	
142.	$(112 \times 5^4) = ?$		$(112 \times 5^4) = 112 \times (10)4 = 112 \times 10^4 = 1120000 = 7000022^4 16$	
143.	It was Sunday on Jan 1, 2006.The day of the week Jan 1, 2010 is		On 31^{st} December, 2005 it was Saturday. Number of odd days from the year 2006 to the year 2009 = (1 + 1 + 2 + 1) = 5 days. ∴ On 31^{st} December 2009, it was Thursday. Thus, on 1^{st} Jan, 2010 it is Friday.	
144.	Today is Monday. After 61 days, it will be:		Each day of the week is repeated after 7 days. So, after 63 days, it will be Monday. ∴ After 61 days, it will be Saturday.	
145.	If 6^{th} March, 2005 is Monday, The day of the week on 6^{th} March, 2004 is		The year 2004 is a leap year. So, it has 2 odd days. But, Feb 2004 not included because we are calculating from March 2004 to March 2005. So it has 1 odd day only. ∴ The day on 6^{th} March, 2005 will be 1	

			day beyond the day on 6^{th} March, 2004. Given that, 6^{th} March, 2005 is Monday. $\therefore$ 6^{th} March, 2004 is Sunday (1 day before to 6^{th} March, 2005).	
146.	The daysin x weeks x days?		x weeks x days = $(7x + x)$ days = $8x$ days.	
147.	On 8^{th} Feb, 2005 it was Tuesday. The day of the week on 8^{th} Feb, 2004 is		The year 2004 is a leap year. It has 2 odd days. $\therefore$ The day on 8^{th} Feb, 2004 is 2 days before the day on 8^{th} Feb, 2005. Hence, this day is Sunday.	
148.	Find the greatest number that will divide 43, 91 and 183 so as to leave the same remainder in each case.		Required number = H.C.F. of (91 - 43), (183 - 91) and (183 - 43) = H.C.F. of 48, 92 and 140 = 4.	
149.	The H.C.F. of two numbers is 23 and the other two factors of their L.C.M. are 13 and 14. The larger of the two numbers is:		Clearly, the numbers are (23 x 13) and (23 x 14). $\therefore$ Larger number = (23 x 14) = 322	
150.	Two trains running in opposite directions cross a man standing on the platform in 27 seconds and 17 seconds respectively and they cross each other in 23 seconds. The ratio of their speeds is:		Let the speeds of the two trains be x m/sec and y m/sec respectively. Then, length of the first train = $27x$ meters, and length of the second train = $17y$ meters. $\therefore \frac{27x + 17y}{x+y} = 23$ $\Rightarrow 27x + 17y = 23x + 23y$ $\Rightarrow 4x = 6y$ $\Rightarrow \frac{x}{y} = \frac{3}{2}$.	

Faculty Team Prepared **Signatures**

1. **Mr.T.Manivel**

HoD