SILIGURI INSTITUTE OF TECHNOLOGY

COMPUTER SCIENCE & ENGINEERING

COURSE FILE

1ST SEM, 3RD YEAR, 2016

PAPER NAME : Design & Analysis of Algorithm

PAPER CODE : CS 501 & CS 591

Course Title/Code: Design and Analysis of Algorithm/CS501 & CS591

Semester:- 1st Year:- 3rd Group:- B

Name of the Faculty: **Sucharita Das**
E-mail : sucharita.das007@gmail.com

Class Schedule:

Day	Monday	Wednesday		Thursday	Friday	Saturday
Timing(B)	**10:00 AM – 10:50 AM**	**12:30 PM -1:20 PM**	**2:10 PM-3:00 PM (Tutorial)**	**2:10 PM-3:00 PM (Tutorial)**	**10:00 AM–10:50 AM**	**12:30 PM -1:20 PM**

Laboratory Schedule:

Day	Monday	Tuesday	Wednesday	Thursday	Friday
Group B1	---	---	---	---	**2:10 pm – 4:40 pm**
Group B2	---	**2:10 pm – 4:40 pm**	---	---	**---**

Hours of Meeting Students: Monday (11:00 AM – 5:00PM) and Wednesday(10:00AM-12:15 PM)/By an Appointment

i) Course Objective:

Students will be able to apply different programming design paradigm to develop new algorithms and also analyze the efficiency of its algorithm.

ii) Course Outcomes:

After completion of this course the students are expected to be able to demonstrate following knowledge, skills and attitudes.

a) The Students will be able to:

Course Outcomes		Targets
CS501.1	**Memorize** the fundamental principles of basic algorithms. (**BT-Level 1**)	60% marks
CS501.2	**Describe** the notion of NP-completeness. (**BT-Level 2**)	60% marks
CS501.3	**Use** the Asymptotic notations as well as Recurrences on simple algorithms, including those algorithms that are using complex loops and recursions. (**BT-Level 3**)	60% marks
CS501.4	**Estimate** the time and space complexity of a given algorithm. (**BT-Level 4**)	60% marks
CS501.5	**Experiment** and analysis on various algorithms on graph data structures as well as basic graph algorithms. (**BT-Level 5**)	60% marks
CS501.6	**Implement** different known algorithms with the help of different programming design paradigm like divide & conquer, greedy method, dynamic programming , backtracking etc. (**BT-Level 6**)	60% marks

b) Once the student has successfully complete this course, he/she must be able to answer the following questions or perform/demonstrate the following:

SN	QUESTION	BT-LEVEL
1.	**What** do you understand by an algorithm?	1
2.	**What** are the analytic issues of an algorithm?	1
3.	**Write** an algorithm to find the **maximum number among three numbers** and also calculate the running time complexity**.**	1
4.	**Write** an algorithm to calculate the **sum of two matrices** and also calculate the running time complexity.	1
5.	**Define Cook's** theorem. Prove that **3-SAT** is NP- Complete.	2
6.	**Find** out the Recurrence relation of recursive **Tower of Hanoi** problem and solve it for the input size n.	3
7.	**Solve** the following recurrence using iteration method. 1. $T(n)=2T(n/2) + O(n)$	3
8.	**Solve** the following recurrence using master method. 1. $T(n)=2T(n/2) + O(n)$ 2. $T(n)=4T(n/2) + O(n)$ 3. $T(n)=T(n/2) + O(n)$	3
9.	**Show** that the following equation is correct: $33n^2 + 4n = \Omega(n^2)$	3
10.	**Solve** $T(n) = aT(n/b) + O(n^k)$ where $a > 1$ and $b \geq 1$.	3
11.	**Find** out the running time complexity of the **Quick sort** algorithm in Best, Worst and Average cases.	4
12.	**Find** out the running time complexity of the **N-Queen** problem.	4
13.	**Implement** adjacent matrix and adjacent list of a given graph and also conclude which representation is better**.**	5
14.	**Implement** graph traversal techniques like **BFS** and **DFS**.	5
15.	**Implement Binary Search** with the help of **Divide & Conquer** strategy.	6
16.	**Implement** shortest path using **Dijkstra's** algorithm with the help of dynamic programming strategy**.**	6

Design & Analysis of Algorithm syllabus [in Chapters]

Code: CS501

Contact: 3L + 1T

CHAPTER-1

Complexity Analysis: [4L]

Time and Space Complexity, Different Asymptotic notations – their mathematical significance

CHAPTER-2

Heap Sort and its complexity [2L]

CHAPTER-3

Divide and Conquer: [3L]

Basic method, use, following case studies with proper analysis.

1) Binary Search.
2) Merge Sort.
3) Quick Sort and their complexity.

CHAPTER-4

Dynamic Programming: [4L]

Basic method, use, following case studies with proper analysis.

1) Matrix Chain Multiplication.
2) All pair shortest paths
    a. Floyd-Warshall Algorithm.
3) Single source shortest path.
    a. Dijkstra's Algorithm.
    b. Bellmanford Algorithm.

CHAPTER-5

Backtracking: [2L]

Basic method, use, following case studies with proper analysis.

1) n queens problem.
2) Graph coloring problem.

CHAPTER-6

Greedy Method: [4L]

Basic method, use, ,following case studies with proper analysis.

1) Knapsack problem.
2) Job sequencing with deadlines.
3) Minimum cost spanning tree
    a. Prim's Algorithm.
    b. Kruskal's Algorithm.

CHAPTER-7

Lower Bound Theory: [1L]

Prove O(nlg(n)) bound for comparison sort .

CHAPTER-8

Disjoint set manipulation: [1L]

Set manipulation algorithm like UNION-FIND, union by rank.

CHAPTER-9

Graph traversal algorithm: [3L]

1) Breadth First Search(BFS)
2) Depth First Search(DFS)
3) Classification of edges - tree, forward, back and cross edges – complexity and comparison

CHAPTER-10

String matching problem: [2L]

Different techniques Naive algorithm, string matching using finite automata, and Knuth, Morris, Pratt (KMP) algorithm with their complexities.

CHAPTER-11

Amortized Analysis: [2L]

Aggregate, Accounting, and Potential Method.

CHAPTER-12

Network Flow: [3L]

Ford Fulkerson algorithm, Max-Flow Min-Cut theorem (Statement and Illustration)

CHAPTER-13

Matrix Manipulation Algorithm: [3L]

Strassen's matrix manipulation algorithm; application of matrix multiplication to solution of simultaneous linear equations using LUP decomposition, Inversion of matrix and Boolean matrix multiplication.

CHAPTER-14

Notion of NP-completeness: [4L]

P class, NP class, NP hard class, NP complete class – their interrelationship, Satisfiability problem, Cook's theorem (Statement only), and Clique decision problem.

CHAPTER-15

Approximation Algorithms:[1L]

Necessity of approximation scheme, performance guarantee, polynomial time approximation schemes, vertex cover problem, travelling salesman problem.

c) Chapter Layout

Chapter No.	Chapter	Lecture Hours	Tutorials	Laboratory hours
Chapter - 1	Complexity Analysis	4 HRS	1	6 HRS
Chapter – 2	Heap Sort and its complexity	2 HRS	1	3 HRS
Chapter – 3	Divide and Conquer	3 HRS	1	3 HRS
Chapter – 4	Dynamic Programming	4 HRS	1	6 HRS
Chapter – 5	Backtracking	2 HRS		3 HRS
Chapter – 6	Greedy Method	4 HRS	1	3 HRS
Chapter – 7	Lower Bound Theory	1 HRS		

Chapter – 8	Disjoint set manipulation	1 HRS	1	
Chapter – 9	Graph traversal algorithm	3 HRS	1	3 HRS
Chapter – 10	String matching problem	2 HRS	1	3 HRS
Chapter – 11	Amortized Analysis	2 HRS		
Chapter – 12	Network Flow:	3 HRS	1	
Chapter – 13	Matrix Manipulation Algorithm	3 HRS	1	
Chapter – 14	Notion of NP-completeness	4 HRS	1	
Chapter - 15	Approximation Algorithms	1 HRS		
Total		39 HRS	11	30 HRS

d) Textbooks:

1. T. H. Cormen, C. E. Leiserson, R. L. Rivest and C. Stein , "Introduction to Algorithms"
2. Aho, J. Hopcroft and J. Ullman "The Design and Analysis of Algorithms" D. E. Knuth "The Art of Computer Programming", Vol. 3
3. Jon Kleiberg and Eva Tardos, "Algorithm Design"

e) Reference Books:

1. K. Mehlhorn , "Data Structures and Algorithms" - Vol. I & Vol. 2.
2. S. Baase "Computer Algorithms"
3. E. Horowitz and Shani "Fundamentals of Computer Algorithms"

f) Evaluation Scheme:

1) THEORY

Evaluation Criteria	Marks
First & Second Internal Exam*	15
Quiz/ Assignments	10
Attendance	5
University Exam	70
Total	100

*Two internal examinations are conducted; based on those two tests, average of them are considered in a scale of 15.

University Grading System:

Grade	Marks
O	90% and above
E	80 – 89.9%
A	70 – 79.9%
B	60 – 69.9%
C	50 – 59.9%
D	40 – 49.9%
F	Below 40%

2) LABORATORY

Evaluation Criteria	Marks
Internal Exam*	40
University Exam	60
Total	100

***** Internal Evaluation will be based on daily lab performance as per the following schedule:

g) Laboratory Evaluation:

Expt. No.	Experiment Name	Schedule	Marks
P1	Experiment on different **Searching** Techniques and also judge the running time complexity. List of Experiments --- **1) Linear Search** **2) Binary Search**	3 HRS	2 + 2
P2	Experiment on different **Sorting** techniques and also judge the running time complexity. List of Experiments --- **3) Merge Sort** **4) Quick Sort**	3 HRS	2 + 2
P3	Experiment on different **Sorting** techniques and also judge the running time complexity. List of Experiments --- **5) Heap Sort** **6) Counting Sort**	3 HRS	2 + 2
P4	Experiment on some **recursion** problems also judge the running time complexity as well as plot the graph. List of Experiments --- **7) Calculate x^y** **8) N^{th} Fibonacci Number** **9) Tower of Hanoi etc and**	3 HRS	1+1+2
P5	Experiment on **Dynamic Programming** algorithm strategy and also judge the running time complexity. **10) Matrix Chain Multiplication.**	3 HRS	4
P6	Experiment on **Dynamic Programming** algorithm strategy and also judge the running time complexity. **11) Floyd's Algorithm**	3 HRS	4
P7	Experiment on **Backtracking** algorithm strategy and also judge the running time complexity. List of Experiments --- **12) 4 Queen** **13) Graph Coloring**	3 HRS	2+2

P8	Experiment on **Minimum Spanning Tree** and also judge the running time complexity. (Any one) List of Experiments --- **14) Prim's Algorithm** **15) Kruskal's Algorithm**	3 HRS	4
P9	Experiment on **Graph Traversal Techniques** and also judge the running time complexity. List of Experiments --- **16) BFS** **17) DFS**	3 HRS	2 + 2
P10	Experiment on **String Matching** Algorithm and also judge the running time complexity. **18) KMP**	3 HRS	4

Overall Course Attainment Target

Attainment Level	Inference	Marks
Attainment Level 1	50% of the students have attained more than the target level of that CO	1
Attainment Level 2	60% of the students have attained more than the target level of that CO	2
Attainment Level 3	70% of the students have attained more than the target level of that CO	3

(70% of university and 30% of the internal exam) will be = **Attainment Level 3**

Target has been set on the basis of last year's performance / result by the students, student quality this year and difficulty level of the course.

h) Mapping of Course Outcomes and Program Outcomes:

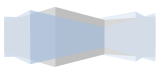
Course Outcomes	Program Outcomes (PO's)												PSOs	
	1	2	3	4	5	6	7	8	9	10	11	12	1	2
CS501.1	1	1	1	-	-	-	-	-	-	-	-	-	1	1
CS501.2	1	2	2	-	-	--	-	-	-	-	-	-	2	-
CS501.3	1	1	2	-	-	-	-	-	-	-	-	-	2	-
CS501.4	2	2	2	-	-	--	-	-	-	-	-	-	2	1
CS501.5	1	2	2	-	-	-	-	-	1	-	-	-	-	2
CS501.6	1	3	2	-	-	-	-	-	1	-	-	-	-	2
CS501	1	2	2	-	-	-	-	-	1	-	-	-	2	2

i) Delivery Methodology:

Outcome	Method	Supporting Tools	Demonstration
CS 501.1	Structured (partially supervised)	Blackboard	Describe the basic algorithm.
CS 501.2	Structured (partially supervised)	Blackboard , PPT , NPTEL	Describe the basic concept of NP Completeness.
CS 501.3	Structured (partially supervised)	Blackboard + C Programming	Calculate Asymptotic notations & Recurrences
CS 501.4	Structured (partially supervised)	Blackboard , NPTEL	Judge the efficiency of a given algorithm.
CS 501.5	Structured (partially supervised)	Blackboard + C Programming	Implement different graph traversal algorithms.
CS 501.6	Structured (partially supervised)	Blackboard + C Programming	Implement basic algorithm with the help of different programming design paradigm.

j) Assessment Methodology:

Assessment Tool	Outcomes						Specific Question/activity aligned to the Outcome
	CS501.1	CS501.2	CS501.3	CS501.4	CS501.5	CS501.6	
FIRST INTERNAL	√	--	√	√	√	√	First Internal Question Paper
SECOND INTERNAL	√	--		√	√	√	Second Internal Question Paper
ASSIGNMENT	√	--	√	√	√	√	First, Second, Third Assignment
QUIZ	√	√	√	√	√	√	Quiz –(Q1, Q2, Q3)
LABORATORY	--	--	√	√	√	√	LAB Assignments

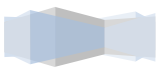


k) A. Weekly Lesson Plan

Week	Lecture	Tutorial	Laboratory	Assignment/Quiz
1	Complexity Analysis.	Tutorial on Linear/Non Linear Data Structures. (TS1)	Review on basic algorithms.	---
2	Heap Sort.	Tutorial on Asymptotic Notations & Recurrences (TS2)	Recursion (P4)	Assignment - I (A1)
3	Binary Search, Merge Sort, Quick Sort.	Tutorial on Heap Sort & Binary Search. (TS3)	Linear & Binary Search (P1)	---
4	Matrix Chain Multiplication. Single Source Shortest Path. (Dijkstra's & Bellman Ford)	Tutorial on Binary Search, Merge Sort & Quick Sort. (TS4)	Sorting (P2)	Assignment - II (A2)
5	All Pair Shortest Path (Floyd's Algorithm). N-Queen, Graph Coloring.	Tutorial on Matrix Chain Multiplication & Shortest Path Problem. (TS5)	Sorting (P3)	Quiz – I (Q1)
6	Knapsack problem. Kruskal's Algorithm. Prim's Algorithm.	Tutorial on Floyd's algorithm. (TS6)	Matrix Chain (P5)	---
7	Job Sequencing with deadline. Lower Bound Theory. Disjoint set manipulation.	Tutorial on Knapsack Problem & Job Sequencing. (TS7)	Floyd's (P6)	Assignment - III (A3)
8	Graph traversal algorithm (BFS & DFS)	Tutorial on MST. (TS8)	N-Queen & Graph Coloring (P7)	Quiz – II (Q2)
9	String Matching.	Tutorial on BFS & DFS. (TS9)	Prim's & Kruskal's (P8)	---
10	Amortized Analysis Approximation Algorithms.	-----	BFS & DFS (P9)	---
11	Network Flow, Ford- Fulkerson algorithm.	-----		---
12	System of Linear Equations Solve by LUP. Strassen's Matrix Multiplication. Matrix Inversion & Boolean Matrix Multiplication.	Tutorial on Network Flow. (TS12)	KMP (P10)	---
13	Notion of NP-completeness.	Tutorial on LUP. (TS10) Tutorial on Matrix Inversion. (TS11)	---	Quiz – III (Q3)

B. Daily Lesson Plan (Repeat format for each chapter)

CHAPTER: 1 Title: **Time & space complexity** Date: **15th July,2016**, Day: **Friday**
CONTENTS Asymptotic notations & other mathematical preliminaries with examples.
Chapter Objectives: They are capable to make a decision what are the actual ways to judge the efficiency of an algorithm. **Broad Objectives of the chapter are:** 1. To able how to judge the efficiency of an algorithm in **worst** case. 2. To able how to judge the efficiency of an algorithm in **best** case. 3. To able how to judge the efficiency of an algorithm in **average** case.
Once the student has completed this topic/ chapter he/she will be able to answer following questions/perform the following activities with Levels of Bloom's Taxonomy: 1. **What** is asymptotic notation? (Level 1) 2. **Different** types of asymptotic notations. (Level 2) 3. **Explain** Big 'O' notation with example. (Level 4) 4. **Explain** Big 'Ω' notation with example. (Level 4) 5. **Explain** Big 'Θ' notation with example. (Level 4) 6. **Prove** that $2n^2 + 5n + 4 = O(n^2)$/ $\Omega(n^2)$/ $\Theta(n^2)$ (Level 5)
HOME WORK: 1. Prove that $3n^2 + 7n = O(n^2)$ (Like Question No. 6) 2. Prove that $3n^2 + 7n = \Omega(n^2)$ (Like Question No. 6) 3. Prove that $3n^2 + 7n = \Theta(n^2)$ (Like Question No. 6) 4. Short notes on asymptotic notations. (Like Question No. 1-5)



CHAPTER: 1

Title: **Time & space complexity**

Date: **16th July,2016**, Day: **Saturday**

CONTENTS

Recursion & iteration, design of recursive algorithms, tower of Hanoi, tail recursion

Chapter Objectives: They are capable to make a decision what are the actual ways to judge the efficiency of an algorithm.

Broad Objectives of the chapter are:

1. To able to write an iterative algorithm.
2. To able to write a recursive algorithm.
3. To able how to judge the efficiency of an iterative algorithm.
4. To able how to judge the efficiency of a recursive algorithm.

Once the student has completed this topic/ chapter he/she will be able to answer following questions/perform the following activities with Levels of Bloom's Taxonomy:

1. How to write an iterative algorithm. (Level 2)
2. How to write recursion algorithm. (Level 2)
3. How to judge the efficiency of an iterative algorithm. (Level 2)
4. How to judge the efficiency of a recursive algorithm. (Level 2)
5. Write an algorithm on Tower of Hanoi problem. (Level 1)
6. What is tail recursion with example? (Level 1)

HOME WORK:

1. Write an iterative algorithm on n^{th} Fibinacci number & calculate the running time complexity. (Like Question No. 1&3)
2. Write a recursive algorithm on n^{th} Fibinacci number & calculate the running time complexity. (Like Question No. 2 & 4)
3. Short notes on Tower of Hanoi problem. (Like Question No. 5,6)

CHAPTER: 1

Title: **Time & space complexity**

Date: **18th July,2016**, Day: **Monday**

CONTENTS

Substitution Method with examples, Iteration Method with examples Master Method with examples.

Chapter Objectives: They are capable to make a decision what are the actual ways to judge the efficiency of an algorithm.

Broad Objectives of the chapter are:

1. They are able, how to judge efficiency of an algorithm using substitution method.
2. They are able, how to judge efficiency of an algorithm using iteration method.
3. They are able, how to judge efficiency of an algorithm using master method.

Once the student has completed this topic/ chapter he/she will be able to answer following questions/perform the following activities with Levels of Bloom's Taxonomy:

1. How to solve substitution method with example. (Level 6)
2. How to solve iteration method with example. (Level 6)
3. How to solve master method with example. (Level 6)

HOME WORK:

1. Solve the following recurrence using substitution method (Like Question No. 1)
    a. $T(n)=2T(n/2) + O(n)$
2. Solve the following recurrence using iteration method (Like Question No. 2)
    a. $T(n)=2T(n/2) + O(n)$
3. Solve the following recurrence using master method (Like Question No. 3)
    a. $T(n)=2T(n/2) + O(n)$
    b. $T(n)=4T(n/2) + O(n)$
    c. $T(n)=T(n/2) + O(n)$
4. Short notes on Recurrences. (Like Question No. 1-3)

CHAPTER: 1

Title: **Time & space complexity**

Date: **20th July,2016**, Day: **Wednesday**

CONTENTS

Different algorithms for a problem, example study – Fibonacci numbers using recursion & iteration with complexity

Chapter Objectives: They are capable to make a decision what are the actual ways to judge the efficiency of an algorithm.

Broad Objectives of the chapter are:

1. They are able to judge the efficiency of an algorithm in **worst** case with case study.
2. They are able to judge the efficiency of an algorithm in **best** case with case study.
3. They are able to judge the efficiency of an algorithm in **average** case with case study.

Once the student has completed this topic/ chapter he/she will be able to answer following questions/perform the following activities with Levels of Bloom's Taxonomy:

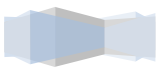
1. How to judge the efficiency of an algorithm in **worst** case with case study. (Level 5)
2. How to judge the efficiency of an algorithm in **best** case with case study. (Level 5)
3. How to judge the efficiency of an algorithm in **average** case with case study. (Level 5)

HOME WORK:

1. Write an iterative algorithm on sum of n numbers & calculate the running time complexity. (Like Question No. 1, 2 & 3)
2. Write a recursive algorithm on sum of n numbers & calculate the running time complexity. (Like Question No. 1, 2 & 3)

LABORATORY EXPERIMENT: (P4)

1. WAP to find out nth Fibonacci number using recursion as well as tail recursion and calculate the running time complexity also plot the curve between certain ranges.
2. WAP to solve Tower of Hanoi problem using recursion.
3. WAP to compute x^y.



Tutorial 1

Title: **Tutorial on Linear / Non Linear Data Structures**

Date: **20th and 21st July,2016**, Day: **Wednesday and Thursday**

CHAPTER: 2

Title: **Heap Sort and its complexity**

Date: **22th July,2016**, Day: **Friday**

CONTENTS

Discuss on Heap, Types of heap, how to create heap with examples.

Chapter Objectives: They are capable to make a heap as well as heap sort and judge the efficiency of this algorithm.

Broad Objectives of the chapter are:

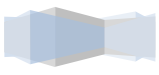
1. They are able to describe what is heap.
2. They are able to know types of heap.
3. They are able to create a heap.

Once the student has completed this topic/ chapter he/she will be able to answer following questions/perform the following activities with Levels of Bloom's Taxonomy:

1. **What** is heap? (Level 1)
2. **Different** types of heap. (Level 2)
3. **Create** a heap for some certain data. (Level 6)

HOME WORK:

1. Create a heap (max/min) with the following data 33, 25, 67, 89, 12, 55, 3, 67. (Like Question No. 3)
2. Short notes on heap. (Like Question No. 1-3)



CHAPTER: 2

Title: **Heap Sort and its complexity**

Date: **25th July,2016**, Day: **Monday**

CONTENTS

Heap sort with example. Analysis of heap as well as heap sort.

Chapter Objectives: They are capable to make a heap as well as heap sort and judge the efficiency of the algorithm.

Broad Objectives of the chapter are:

1. They are able to explain heap sort algorithm with some example.
2. They are able to solve the efficiency of heap sort algorithm.

Once the student has completed this topic/ chapter he/she will be able to answer following questions/perform the following activities with Levels of Bloom's Taxonomy:

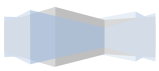
1. Write the heap sort algorithm with example? (Level 1)
2. Short notes on heap sort. (Level 2)
3. What is the time complexity of a heap? (Level 1)
4. Establish the time complexity of a heap sort. (Level 4)

HOME WORK:

1. Create a heap (max/min) with the following data 33, 25, 67, 89, 12, 55, 3, 67. And sort the data in ascending and descending order. (Like Question No. 1)
2. Short notes on heap sort. (Like Question No. 1-4)

LABORATORY EXPERIMENT:

1. WAP to implement Heap sort. Estimate the running time complexity.



CHAPTER: 3

Title: **Divide & Conquer**

Date: **27th July,2016**, Day: **Wednesday**

CONTENTS

Basic idea on divide & conquer (D&C) method: case study - Binary Search

Chapter Objectives: They are capable to make an algorithm on the basis of D&C strategy and judge the efficiency of the algorithm.

Broad Objectives of the chapter are:

1. They are able to explain D&C algorithm strategy with some example.
2. They are able to explain binary search algorithm with some example.
3. They are able to solve the efficiency of binary search algorithm.

Once the student has completed this topic/ chapter he/she will be able to answer following questions/perform the following activities with Levels of Bloom's Taxonomy:

1. Write the binary search algorithm with example? (Level 1)
2. What is BST? (Level 1)
3. What is the time complexity of binary search algorithm? (Level 4)
4. Establish the time complexity of a binary search algorithm. (Level 4)

HOME WORK:

1. Create a binary search tree (BST) with the following data 33, 25, 67, 89, 12, 55, 3, 67. (like question no. 2)
2. Short notes on binary search technique. (like question no. 1,3,4)
3. Explain binary search algorithm with an example. (like question no. 1)

LABORATORY EXPERIMENT:

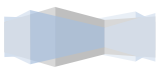
1. WAP to implement binary search iterative as well as recursive method. Establish the running time complexity for both the cases.

Tutorial 2

Title: **Asymptotic Notations & Recurrences**

Date: **27th and 28th July,2016**, Day: **Wednesday and Thursday**

1. Prove that $3n^2 + 7n = O(n^2)$
2. Prove that $3n^2 + 7n = \Omega(n^2)$
3. Prove that $3n^2 + 7n = \Theta(n^2)$
4. Solve the following recurrence using substitution method.
    a. $T(n)=2T(n/2) + O(n)$
5. Solve the following recurrence using iteration method.
    a. $T(n)=2T(n/2) + O(n)$
6. Solve the following recurrence using master method.
    a. $T(n)=2T(n/2) + O(n)$
    b. $T(n)=4T(n/2) + O(n)$
    c. $T(n)=T(n/2) + O(n)$



CHAPTER: 3

Title: **Divide & Conquer**

Date: **29th July,2016**, Day: **Friday**

CONTENTS

Divide & conquer method: (contd.) case study - Merge Sort .

Chapter Objectives: They are capable to make an algorithm on the basis of D&C strategy and judge the efficiency of the algorithm.

Broad Objectives of the chapter are:

1. They are able to explain merge sort algorithm with some example.
2. They are able to solve the efficiency of merge sort algorithm.

Once the student has completed this topic/ chapter he/she will be able to answer following questions/perform the following activities with Levels of Bloom's Taxonomy:

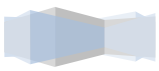
1. Write the merge sort algorithm with example? (Level 1)
2. What is the time complexity of merge sort algorithm? (Level 4)
3. Establish the time complexity of a merge sort algorithm. (Level 4)

HOME WORK:

1. Short notes on merge sort. (like question no 1-3)
2. Explain merge sort algorithm with an example. (like question no. 1)

LABORATORY EXPERIMENT:

1. WAP to implement merge sort using divide & conquer strategy. Establish the running time complexity.



CHAPTER: 3

Title: **Divide & Conquer**

Date: **30th July,2016**, Day: **Saturday**

CONTENTS

Divide & conquer method: (contd.) case study - Quick Sort .

Chapter Objectives: They are capable to make an algorithm on the basis of D&C strategy and judge the efficiency of the algorithm.

Broad Objectives of the chapter are:

1. They are able to explain quick sort algorithm with some example.
2. They are able to solve the efficiency of quick sort algorithm.

Once the student has completed this topic/ chapter he/she will be able to answer following questions/perform the following activities with Levels of Bloom's Taxonomy:

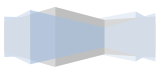
1. Write the quick sort algorithm with example? (Level 1)
2. What is the time complexity of quick sort algorithm? (Level 4)
3. Establish the time complexity of a quick sort algorithm. (Level 4)

HOME WORK:

1. Short notes on quick sort. (like question no 1-3)
2. Explain quick sort algorithm with an example. (like question no. 1)

LABORATORY EXPERIMENT:

1. WAP to implement quick sort using divide & conquer strategy. Establish the running time complexity.



CHAPTER: 4

Title: **Dynamic Programming**

Date: **1st August,2016**, Day: **Monday**

CONTENTS

Dynamic programming: case study –Matrix Chain Multiplication with example

Chapter Objectives: They are capable to make an algorithm on the basis of Dynamic Programming strategy and judge the efficiency of the algorithm.

Broad Objectives of the chapter are:

1. They are able to explain matrix chain multiplication algorithm with some example.
2. They are able to solve the efficiency of matrix chain multiplication algorithm.

Once the student has completed this topic/ chapter he/she will be able to answer following questions/perform the following activities with Levels of Bloom's Taxonomy:

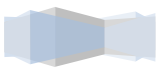
1. Write the matrix chain multiplication algorithm with example? (Level 1)
2. Write the optimal parenthesis algorithm with example? (Level 1)
3. What is the time complexity of matrix chain multiplication algorithm? (Level 1)
4. Establish the time complexity of a matrix chain multiplication algorithm. (Level 4)

HOME WORK:

1. Short notes on matrix chain multiplication. (like question no 1,3,4)
2. Find an optimal parenthesization of a matrix-chain product whose sequence of dimensions is⟨5,10,3,12,5⟩. (like question no. 1)

LABORATORY EXPERIMENT:

1. WAP to multiply a chain of matrices optimally whose sequence of dimensions is <10, 20, 50, 1, 100>.



CHAPTER: 4

Title: **Dynamic Programming**

Date: **3rd August,2016**, Day: **Wednesday**

CONTENTS

Dynamic programming: case study – Single Source shortest paths with example.
(Dijkstra's Algorithm)

Chapter Objectives: They are capable to make an algorithm on the basis of Dynamic Programming strategy and judge the efficiency of the algorithm.

Broad Objectives of the chapter are:

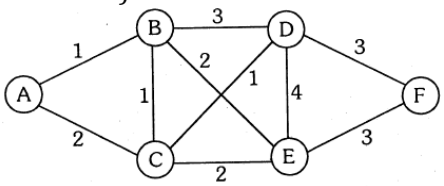
1. They are able to clarify the actual need of single source shortest path problem with an example.
2. They are able to explain Dijkstra's algorithm with some example.
3. They are able to solve the efficiency of Dijkstra's algorithm with different data structure.

Once the student has completed this topic/ chapter he/she will be able to answer following questions/perform the following activities with Levels of Bloom's Taxonomy:

1. Write the Dijkstra's algorithm of single source shortest path problem with an example? (Level 1)
2. Establish the time complexity of a Dijkstra's algorithm of single source shortest path algorithm. (Level 4)

HOME WORK:

1. Short notes on single source shortest path problem. (like question no 1,2)
2. Find out the shortest path between Vertex 'A' to Vertex 'F' using Dijkstr's algorithm where Vertex 'A' is the start Vertex. (like question no. 1)



LABORATORY EXPERIMENT:

1. WAP to implement Dijkstra's algorithm for single source shortest path. (**if required**)

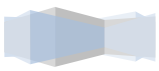
Tutorial: 3

Title: **Heap Sort**

Date: **3rd and 4th August,2016**, Day: **Wednesday and Thursday**

1) Create a binary heap (max/min) with the following data 33, 25, 67, 89, 12, 55, 3, 67. And sort the data in ascending and descending order.

2) Create a Fibonacci heap with the following data 33, 25, 67, 89, 12, 55, 3, 67.



CHAPTER: 4

Title: **Dynamic Programming**

Date: **5th August,2016**, Day: **Friday**

CONTENTS

Dynamic programming: case study – Single Source shortest paths with example.
(Bellman ford Algorithm)

Chapter Objectives: They are capable to make an algorithm on the basis of Dynamic Programming strategy and judge the efficiency of the algorithm.

Broad Objectives of the chapter are:

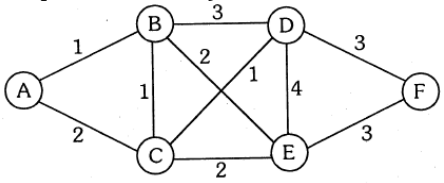
1. They are able to clarify the actual need of single source shortest path problem with an example.
2. They are able to explain Bellman ford algorithm with some example.
3. They are able to solve the efficiency of Bellman ford algorithm with different data structure.

Once the student has completed this topic/ chapter he/she will be able to answer following questions/perform the following activities with Levels of Bloom's Taxonomy:

1. Write the Bellman ford algorithm of single source shortest path problem with an example? (Level 1)
2. Establish the time complexity of a Bellman ford algorithm of single source shortest path algorithm. (Level 4)
3. Compare between Dijkstra's and Bellman ford algorithm.(Level 4)

HOME WORK:

1. Short notes on single source shortest path problem. (like question no 1,2)
2. Find out the shortest path between Vertex 'A' to Vertex 'F' using Bellman ford algorithm where Vertex 'A' is the start Vertex. (like question no. 1)



3. Write some difference between Dijkstra's and Bellman ford algorithm.

LABORATORY EXPERIMENT:

2. WAP to implement Dijkstra's algorithm for single source shortest path. (**if required**)

CHAPTER: 4

Title: **Dynamic Programming**

Date: **6th August,2016**, Day: **Saturday**

CONTENTS

Dynamic programming: case study – All pair shortest paths with example. (Floyd's Algorithm)

Chapter Objectives: They are capable to make an algorithm on the basis of Dynamic Programming strategy and judge the efficiency of the algorithm.

Broad Objectives of the chapter are:

1. They are able to clarify the actual need of all pair shortest path problem with an example.
2. They are able to explain Floyd's algorithm with some example.
3. They are able to solve the efficiency of Floyd's algorithm.

Once the student has completed this topic/ chapter he/she will be able to answer following questions/perform the following activities with Levels of Bloom's Taxonomy:

1. Write the Floyd's algorithm for all pair shortest path problem with an example? (Level 1)
2. Establish the time complexity of a Floyd's algorithm of single source shortest path algorithm. (Level 4)

HOME WORK:

1. Short notes on all pair shortest path problem. (like question no 1,2)
2. Find out the shortest path between all pair using Floyd's algorithm. (like question no. 1)

0	7	5	∞
∞	0	7	6
∞	∞	0	∞
4	1	11	0

3. Write some difference between single source and all pair shortest path..

LABORATORY EXPERIMENT:

1. WAP to find the shortest path between all pairs of vertices of a given graph using Floyd's algorithm.

0	7	5	∞
∞	0	7	6
∞	∞	0	∞
4	1	11	0

CHAPTER: 5

Title: **Backtracking**

Date: **8th August,2016**, Day: **Monday**

CONTENTS

Basic idea on backtracking strategy. Case study – n queen problem

Chapter Objectives: They are capable to make an algorithm on the basis of Backtracking strategy and judge the efficiency of the algorithm.

Broad Objectives of the chapter are:

1. They are able to clarify the actual need of backtracking strategy.
2. They are able to explain 'n' queen problem with an example.
3. They are able to solve the efficiency of 'n' queen problem.
4. They are able to know the actual need of state space tree of n queen problem.

Once the student has completed this topic/ chapter he/she will be able to answer following questions/perform the following activities with Levels of Bloom's Taxonomy:

1. Write the short notes on 'n' queen problem. (level 1)
2. Write the 'n' queen algorithm with an example? (Level 1)
3. Establish the time complexity of 'n' queen algorithm. (Level 4)
4. How to draw the state space tree? (Level 2)

HOME WORK:

1. Short notes on n queen problem. (like question no 1,2,3)
2. Find out the one solution for the given 4 queen problem. (like question no. 2)

-	Q	-	-
-	-	-	-
-	-	-	-
-	-	-	-

3. Draw the state space tree for 4 queen problem. (like question no. 4)

LABORATORY EXPERIMENT:

1. WAP to implement 4-Queen problem using backtracking strategy.

CHAPTER: 5

Title: **Backtracking**

Date: **10th August,2016**, Day: **Wednesday**

CONTENTS

Basic idea on backtracking strategy. Case study – graph coloring problem

Chapter Objectives: They are capable to make an algorithm on the basis of Backtracking strategy and judge the efficiency of the algorithm.

Broad Objectives of the chapter are:

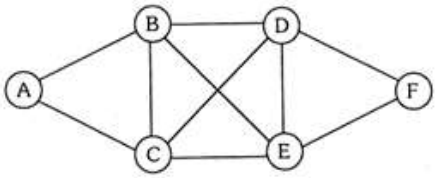
1. They are able to explain graph coloring problem with an example.
2. They are able to solve the efficiency of graph coloring algorithm.
3. They are able to know the actual need of state space tree of graph coloring problem.

Once the student has completed this topic/ chapter he/she will be able to answer following questions/perform the following activities with Levels of Bloom's Taxonomy:

1. Write the short notes on graph coloring problem. (level 1)
2. Write the 'graph coloring algorithm with an example? (Level 1)
3. Establish the time complexity of graph coloring algorithm. (Level 4)
4. How to draw the state space tree? (Level 2)

HOME WORK:

1. Short notes on graph coloring problem. (like question no 1,2,3)
2. Find how many distinct colors are required for the given graph.(like question no. 2)



3. Draw the state space tree for 3-coloring problem. (like question no. 4)

LABORATORY EXPERIMENT:

1. WAP to implement graph coloring problem using backtracking strategy.

Tutorial: 4

Title: **Binary Search, Merge Sort & Quick Sort**

Date: **10th and 11th August,2016**, Day: **Wednesday and Thursday**

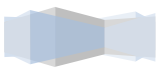
1) Create a binary search tree (BST) with the following data 33, 25, 67, 89, 12, 55, 3, 67 and also find out the results in different traversal techniques

2) Illustrate the operation of PARTITION on the following sequence of keys.
   a. 2, 3, 18, 17, 5, 1

3) Show how Quick sort works for the following sequence of keys.
   a. 2, 3, 18, 17, 5, 1

4) Use Merge sort algorithm to sort the following elements.
   a. 15, 10, 5, 20, 25, 30, 40, 35

5) Show that merging two sorted sequences S_1 and S_2 takes $O(n_1 + n_2)$ time, where n_1 is the size of S_1 and n_2 is the size of S_2.



CHAPTER: 6

Title: **Greedy Method**

Date: **12th August,2016**, Day: **Friday**

CONTENTS

Basic idea on Greedy Method: case study - Knapsack Problem with example.

Chapter Objectives: They are capable to make an algorithm on the basis of Greedy strategy and judge the efficiency of the algorithm.

Broad Objectives of the chapter are:

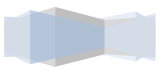
1. They are able to clarify the actual need of greedy strategy with an example.
2. They are able to explain greedy knapsack problem with an example.
3. They are able to solve the efficiency of greedy knapsack algorithm with different data structure.

Once the student has completed this topic/ chapter he/she will be able to answer following questions/perform the following activities with Levels of Bloom's Taxonomy:

1. Write the greedy knapsack algorithm with an example? (Level 1)
2. Establish the time complexity of a greedy knapsack algorithm. (Level 4)

HOME WORK:

1. Short notes on greedy knapsack problem. (like question no 1,2)
2. Find an optimal solution to the knapsack instance n=7, m=15, (v1, v2, v3, . . , v7) = (10, 5, 15, 7, 6, 18, 3), and (w1, w2, w3, . . . , w7) = (2, 3, 5, 7, 1, 4, 1).(like question no 1)



CHAPTER: 6

Title: **Greedy Method**

Date: **17th August,2016**, Day: **Wednesday**

CONTENTS

Greedy Method: Minimum spanning trees(MST): Kruskal's algorithm with example.

Topic/Unit/Chapter Objectives: They are capable to make an algorithm on the basis of Greedy strategy and judge the efficiency of the algorithm.

Broad Objectives of the chapter/topic are:

1. They are able to explain Kruskal's algorithm with an example.
2. They are able to solve the efficiency of Kruskal's algorithm with set data structures.

Once the student has completed this topic/ chapter he/she will be able to answer following questions/perform the following activities with Levels of Bloom's Taxonomy:

1. Write the Kruskal's algorithm with an example? (Level 1)
2. Establish the time complexity of Kruskal's algorithm. (Level 4)

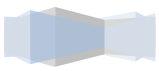
HOME WORK:

1. Short notes on Kruskal's algorithm. (like question no 1,2)
2. Find out the minimum spanning tree for the following graph (adjacent matrix with weight) using Kruskal's algorithm.(like question no 1)

-	1	∞	4	∞	∞	∞
1	-	2	6	4	∞	∞
∞	2	-	∞	5	6	∞
4	6	∞	-	3	∞	4
∞	4	5	3	-	8	7
∞	∞	6	∞	8	-	3
∞	∞	∞	4	7	3	-

LABORATORY EXPERIMENT:

1. WAP to implement MST of a given graph using Kruskal's algorithm. Check your program on a graph whose length matrix is given in (Home Work Question no 2)



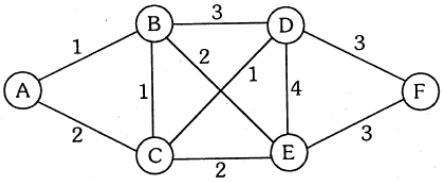
Tutorial: 5

Title: **Matrix Chain Multiplication & Shortest Path Problem**

Date: **17th and 18th August,2016**, Day: **Wednesday and Thursday**

1) Find an optimal parenthesization of a matrix-chain product whose sequence of dimensions is⟨5,10,3,12,5⟩.

2) Find out the shortest path between Vertex 'A' to Vertex 'F' using Dijkstr's / Bellman Ford algorithm where Vertex 'A' is the start Vertex.



CHAPTER: 6

Title: **Greedy Method**

Date: **19th August,2016**, Day: **Friday**

CONTENTS

Greedy Method: Minimum spanning trees(MST): Prim's algorithm with example.

Chapter Objectives: They are capable to make an algorithm on the basis of Greedy strategy and judge the efficiency of the algorithm.

Broad Objectives of the chapter are:

1. They are able to clarify minimum spanning tree with an example.
2. They are able to explain Prim's algorithm with an example.
3. They are able to solve the efficiency of Prim's algorithm with different data structures.

Once the student has completed this topic/ chapter he/she will be able to answer following questions/perform the following activities with Levels of Bloom's Taxonomy:

1. Write the Prim's algorithm with an example? (Level 1)
2. Establish the time complexity of Prim's algorithm. (Level 4)
3. What is minimum spanning tree with an example.(Level 2)

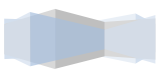
HOME WORK:

1. Short notes on minimum spanning tree. (like question no 1,2)
2. Find out the minimum spanning tree for the following graph (adjacent matrix with weight) using Prim's algorithm. (like question no 1)

-	1	∞	4	∞	∞	∞
1	-	2	6	4	∞	∞
∞	2	-	∞	5	6	∞
4	6	∞	-	3	∞	4
∞	4	5	3	-	8	7
∞	∞	6	∞	8	-	3
∞	∞	∞	4	7	3	-

LABORATORY EXPERIMENT:

2. WAP to implement MST of a given graph using Prim's algorithm. Check your program on a graph whose length matrix is given in (Home Work Question no 2)



CHAPTER: 6

Title: **Greedy Method**

Date: **20th August,2016**, Day: **Saturday**

CONTENTS

Greedy Method: case study - Job Sequencing with Deadline with example.

Chapter Objectives: They are capable to make an algorithm on the basis of Greedy strategy and judge the efficiency of the algorithm.

Broad Objectives of the chapter are:

1. They are able to clarify the actual need of job sequencing with deadline with an example.
2. They are able to explain job sequencing problem with an example.
3. They are able to solve the efficiency of job sequencing with deadline.

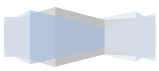
Once the student has completed this topic/ chapter he/she will be able to answer following questions/perform the following activities with Levels of Bloom's Taxonomy:

1. Write the job sequencing with deadline algorithm with an example? (Level 1)
2. Establish the time complexity of job sequencing with deadline algorithm. (Level 4)

HOME WORK:

1. Short notes on job sequencing with deadline problem. (like question no 1,2)
2. Using greedy strategy, schedule the following jobs within deadline so as to maximize the profit. Deadline and profits are mentioned as follows. (like question no 1)

Job i	1	2	3	4
Deadline d_i	3	2	3	1
Profit g_i	9	7	7	2



CHAPTER: 7

Title: **Lower Bound Theory**

Date: **22nd August,2016**, Day: **Monday**

CONTENTS

Lower bound theory, necessity of lower bounds. Lower bound theory: lower bound of the sorting problem with example.

Topic/Unit/Chapter Objectives: They are capable to judge the lower bound of an algorithm.

Broad Objectives of the chapter/topic are:

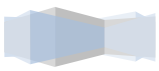
1. They are able to explain Lower bound theory.
2. They are able to clarify the lower bound of the sorting problem.

Once the student has completed this topic/ chapter he/she will be able to answer following questions/perform the following activities with Levels of Bloom's Taxonomy:

1. Justify what is the lower bound of an algorithm? (Level 5)
2. How to design decision trees for sorting algorithms. (Level 6)
3. Establish the lower bound for worst case of comparison based sorting technique. (Level 5)

HOME WORK:

1. Short notes lower bound theory. (like question no 1,2,3)
2. Draw the decision tree for a sorting algorithm where number of element is 4. (like question no 2)



CHAPTER: 8

Title: **Disjoint set manipulation**

Date: **24th August,2016**, Day: **Wednesday**

CONTENTS

Disjoint set manipulation: UNION-FIND, Union by Rank, Path Compression with example.

Topic/Unit/Chapter Objectives: They are competent how to manipulate disjoint sets data structure.

Broad Objectives of the chapter/topic are:

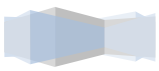
1. They are able to explain disjoint set data structure.
2. They are able to explain Union, Find algorithms.
3. They are able to know path compression with example.

Once the student has completed this topic/ chapter he/she will be able to answer following questions/perform the following activities with Levels of Bloom's Taxonomy:

1. Write Union, Find algorithm with an example? (Level 2)
2. What is path compression, explain with an example. (Level 1)
3. Establish the running time complexity for Union, Find algorithms. (Level 5)
4. Prove the theorem every node has rank at most $\lfloor \lg(n) \rfloor$. (Level 5)
5. Prove the lemma for all tree roots x, size(x) $\geq 2^{rank[x]}$. (level 5)

HOME WORK:

1. Short notes on disjoint set manipulation. (like question no 1,2,3)



Tutorial: 6

Title: **All pair Shortest Path Problem**

Date: **24th August and 1st September,2016**, Day: **Wednesday and Thursday**

1) Find out the shortest path between all pair using Floyd's algorithm. (like question no. 1)

0	7	5	∞
∞	0	7	6
∞	∞	0	∞
4	1	11	0

2) Find out the total number of operations for the above problem.

CHAPTER: 9

Title: **Graph traversal algorithm**

Date: **26th August,2016**, Day: **Friday**

CONTENTS

Graphs, properties of graphs, representation of graphs with examples.

Chapter Objectives: They are capable to make an algorithm and also explain properties of a graph on the basis of graph traversal techniques and judge the efficiency of the algorithm.

Broad Objectives of the chapter are:

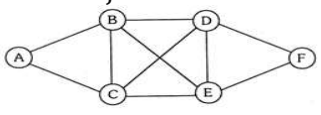
1. They are able to clarify the actual need of graph and also explain graph properties.
2. They are able how to represent a graph in computer memory.

Once the student has completed this topic/ chapter he/she will be able to answer following questions/perform the following activities with Levels of Bloom's Taxonomy:

1. Write the graph representation algorithm with an example? (Level 1)
2. Establish the time complexity of graph representation. (Level 4)

HOME WORK:

1. Short notes on graph. (like question no 1,2)
2. Write the adjacent matrix and adjacent list for the given graph.



3. Prove that a complete graph has at least n(n-1)/2 number of edges.

CHAPTER: 9

Title: **Graph traversal algorithm**

Date: **29th August,2016**, Day: **Monday**

CONTENTS

Graph traversal algorithms: BFS with example.

Chapter Objectives: They are capable to make an algorithm and also explain properties of a graph on the basis of graph traversal techniques and judge the efficiency of the algorithm.

Broad Objectives of the chapter are:

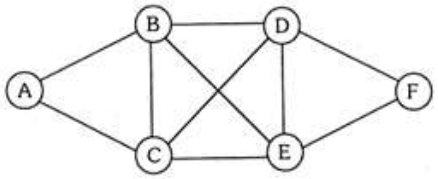
1. They are able to clarify the BFS algorithm with an example.
2. They are able to solve the efficiency of BFS algorithm.

Once the student has completed this topic/ chapter he/she will be able to answer following questions/perform the following activities with Levels of Bloom's Taxonomy:

1. Write the BFS algorithm with an example? (Level 1)
2. Establish the time complexity of BFS algorithm. (Level 4)

HOME WORK:

1. Short notes on BFS. (like question no 1,2)
2. Find out the BFS tree for the given graph. (like question no. 1)



3. Establish the running time for the BFS algorithm.(like question no. 2)

LABORATORY EXPERIMENT:

1. WAP to implement BFS on a given graph(in Home Work Question no 2) where the graph is represented as a adjacent list.

CHAPTER: 9

Title: **Graph Traversal Algorithm**

Date: **31st August,2016**, Day: **Wednesday**

CONTENTS

Graph traversal algorithms: DFS with example.

Chapter Objectives: They are capable to make an algorithm and also explain properties of a graph on the basis of graph traversal techniques and judge the efficiency of the algorithm.

Broad Objectives of the chapter are:

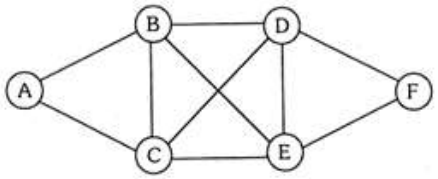
1. They are able to clarify the DFS algorithm with an example.
2. They are able to solve the efficiency of DFS algorithm.

Once the student has completed this topic/ chapter he/she will be able to answer following questions/perform the following activities with Levels of Bloom's Taxonomy:

1. Write the DFS algorithm with an example? (Level 1)
2. Establish the time complexity of DFS algorithm. (Level 4)
3. Comparative study between DFS and BFS.

HOME WORK:

1. Short notes on DFS. (like question no 1,2)
2. Find out the DFS tree for the given graph. (like question no. 1)



3. Establish the running time for the DFS algorithm.(like question no. 2)
4. Difference between DFS and BFS.(like question no. 3).
5. Short notes on Graph traversal techniques. (like question 1,2,3)

LABORATORY EXPERIMENT:

1. WAP to implement DFS on a given graph (in Home Work Question no 2) where the graph is represented as a adjacent list.

Tutorial: 7

Title: Knapsack Problem

Date: **31st August and 8th September,2016**, Day: **Wednesday and Thursday**

1) Find an optimal solution to the knapsack instance n=7, m=15, (v1, v2, v3, . . , v7) = (10, 5, 15, 7, 6, 18, 3), and (w1, w2, w3, . . . , w7) = (2, 3, 5, 7, 1, 4, 1).

CHAPTER: 9

Title: **Graph traversal algorithm**

Date: **2nd September, 2016**, Day: **Friday**

CONTENTS

Graph traversal algorithms: Classification of edges with example.

Chapter Objectives: They are capable to make an algorithm and also explain properties of a graph on the basis of graph traversal techniques and judge the efficiency of the algorithm.

Broad Objectives of the chapter are:

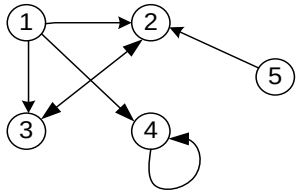
1. They are able to classify the edges in DFS tree.
2. They are able to know different types of edges.

Once the student has completed this topic/ chapter he/she will be able to answer following questions/perform the following activities with Levels of Bloom's Taxonomy:

1. Write the definition of different types of edges with examples. (Level 1)
2. How to classify the edges give an example. (Level 3)

HOME WORK:

1. Find out the DFS tree and classified the edges for the following Graph where start vertex is '1'.



CHAPTER: 10

Title: **String Matching**

Date: **3rd September, 2016**, Day: **Saturday**

CONTENTS

Basic idea on String Matching algorithm: naïve string matching algorithm

Topic/Unit/Chapter Objectives: They are capable to make an algorithm on the basis of string matching and judge the efficiency of the algorithm.

Broad Objectives of the chapter/topic are:

1. They are able to explain what is string algorithm matching with an example.
2. They are able to explain naïve string matching algorithm with example.
3. They are able to find out the running time complexity of naïve string matching algorithm.

Once the student has completed this topic/ chapter he/she will be able to answer following questions/perform the following activities with Levels of Bloom's Taxonomy:

1. Write the naïve string matching algorithm with an example. (Level 2)
2. Establish the running time complexity of naïve string matching algorithm. (Level 4)

CHAPTER: 10

Title: **String Matching**

Date: **5th September, 2016**, Day: **Monday**

CONTENTS

Basic idea on String Matching algorithm: Knuth-Moris-Prat [KMP] string matching algorithm with example.

Topic/Unit/Chapter Objectives: They are capable to make an algorithm on the basis of string matching and judge the efficiency of the algorithm.

Broad Objectives of the chapter/topic are:

1. They are able to explain KMP algorithm with example.
2. They are able to find out the running time complexity of KMP algorithm.
3. They are able to know what is suffix and prefix with example.
4. They are able to know how to calculate the prefix function.

Once the student has completed this topic/ chapter he/she will be able to answer following questions/perform the following activities with Levels of Bloom's Taxonomy:

1. Write the KMP algorithm with an example. (Level 2)
2. Establish the running time complexity of KMP algorithm. (Level 4)
3. Give an example on prefix calculation.(Level 3)

HOME WORK:

1. Write the KMP algorithm with an example.(like question no. 1)

2. Compute the prefix function π for the pattern ababbabbabbababbabb where the alphabet is Σ = {a,b}. (like question no. 3)

LABORATORY EXPERIMENT:

1. WAP to implement KMP algorithm for pattern matching.

CHAPTER: 11

Title: **Amortized Analysis**

Date: **7th September, 2016**, Day: **Wednesday**

CONTENTS

Discuss short notes on Amortized Analysis.

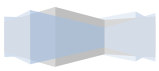
Chapter Objectives: They are capable to judge the efficiency of the algorithm on average running time over per operation cost.

Broad Objectives of the chapter are:

1. They are able to explain average running time per operation cost.
2. They are able to know the actual meaning of amortized analysis.

Once the student has completed this topic/ chapter he/she will be able to answer following questions/perform the following activities with Levels of Bloom's Taxonomy:

1. What is the actual meaning of amortized analysis? (Level 2)



Tutorial: 8

Title: Minimum Spanning Tree

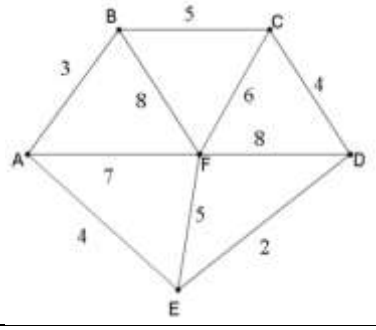
Date: **7th and 15th September,2016**, Day: **Wednesday and Thursday**

1) Find out the minimum spanning tree for the following graphs (adjacent matrix with weight) using Prim's / Krukal's algorithm.

i)

-	1	∞	4	∞	∞	∞
1	-	2	6	4	∞	∞
∞	2	-	∞	5	6	∞
4	6	∞	-	3	∞	4
∞	4	5	3	-	8	7
∞	∞	6	∞	8	-	3
∞	∞	∞	4	7	3	-

ii)



CHAPTER: 11

Title: **Amortized Analysis**

Date: **9th September, 2016**, Day: **Friday**

CONTENTS

Different techniques used in Amortized Analysis.

Chapter Objectives: They are capable to judge the efficiency of the algorithm on average running time over per operation cost.

Broad Objectives of the chapter are:

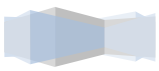
1. They are able to classify different techniques used in amortized analysis.
2. They are able to explain different techniques with an example.

Once the student has completed this topic/ chapter he/she will be able to answer following questions/perform the following activities with Levels of Bloom's Taxonomy:

1. Describe aggregate analysis with an example? (Level 2)
2. Describe accounting method with an example? (Level 2)
3. Describe potential method with an example? (Level 2)

HOME WORK:

1. Write short notes on Amortized analysis.(like question no. 1,2,3)



CHAPTER: 15

Title: **Approximation Algorithm**

Date: **14th September, 2016**, Day: **Wednesday**

CONTENTS

Approximation algorithms: Only Short Notes.

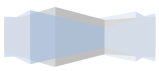
Chapter Objectives: They are capable to understand the notion of Approximation algorithm.

Broad Objectives of the chapter are:

1. They are able to explain actual meaning of approximation algorithm.

Once the student has completed this topic/ chapter he/she will be able to answer following questions/perform the following activities with Levels of Bloom's Taxonomy:

1. Short notes on Approximation algorithm. (Level 4)

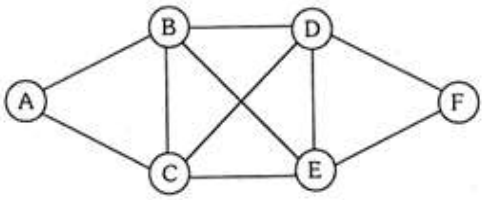


Tutorial: 9

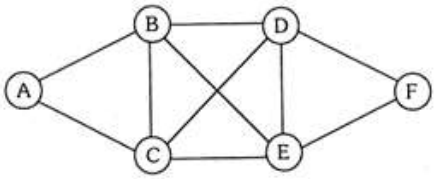
Title: **BFS & DFS**

Date: **14th and 22nd September,2016**, Day: **Wednesday and Thursday**

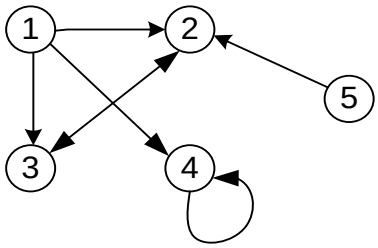
1) Find out the BFS tree for the given graph.

1. 

2) Find out the DFS tree for the given graph.

1. 

3) Find out the DFS tree and classified the edges for the following Graph where start vertex is '1'.



CHAPTER: 12

Title: **Network Flow**

Date: **16th September, 2016**, Day: **Friday**

CONTENTS

Basic idea on Network Flow with examples.

Chapter Objectives: They are capable to explain network flow and also measure the total flow of a network.

Broad Objectives of the chapter are:

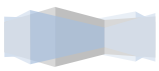
1. They are able to explain about flow networks.
2. They are able to explain certain properties of a flow network.

Once the student has completed this topic/ chapter he/she will be able to answer following questions/perform the following activities with Levels of Bloom's Taxonomy:

1. What is maximum flow network?
2. Describe three properties of a flow network with examples.

HOME WORK:

1. Write short notes on flow network.(like question no. 1,2)
2. What is the meaning of maximal flow problem?(like question no. 1)



CHAPTER: 12

Title: **Network Flow**

Date: **17th September, 2016**, Day: **Saturday**

CONTENTS

Ford Fulkerson Algorithm with example.

Chapter Objectives: They are capable to explain network flow and also measure the total flow of a network.

Broad Objectives of the chapter are:

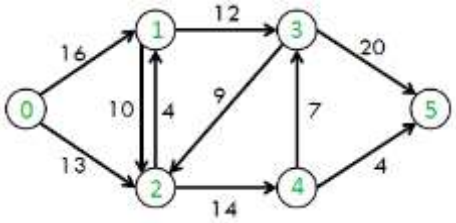
1. They are able to explain Residual network, Augmenting path, Cuts.
2. They are able to explain Ford Fulkerson algorithm of a flow network.

Once the student has completed this topic/ chapter he/she will be able to answer following questions/perform the following activities with Levels of Bloom's Taxonomy:

1. Explain Residual network, augmenting path and cuts with example.(Level 4)
2. Write Ford Fulkerson algorithm on network flow with example. (Level 2)

HOME WORK:

1. Short notes on Ford Fulkerson algorithm.
2. Find out the total flow for the given network using Ford Fulkerson.



CHAPTER: 12

Title: **Network Flow**

Date: **19th September, 2016**, Day: **Monday**

CONTENTS

Discuss on Max Flow Min cut Theorem and illustrate some examples.

Chapter Objectives: They are capable to explain network flow and also measure the total flow of a network.

Broad Objectives of the chapter are:

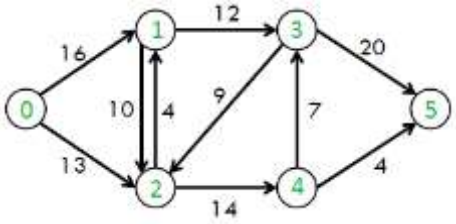
1. They are able to explain Residual network, Augmenting path, Cuts.
2. They are able to explain Max flow Min cut theorem with example.

Once the student has completed this topic/ chapter he/she will be able to answer following questions/perform the following activities with Levels of Bloom's Taxonomy:

1. Explain Residual network, augmenting path and cuts with example.(Level 4)
2. Write Max flow Min cut theorem with example. (Level 3)

HOME WORK:

1. State Max flow Min cut theorem with example.
2. Use the Ford-Fulkerson algorithm to find the maximum flow for the following network. (Source : 0 & Sink: 5) and also find the cuts.



CHAPTER: 13

Title: **Matrix Manipulation Algorithm**

Date: **21st September, 2016**, Day: **Wednesday**

CONTENTS

Algorithms for solution of simultaneous equations using LUP decomposition.

Chapter Objectives: They are capable to make an algorithm on the basis of matrix manipulation and judge the efficiency of the algorithm.

Broad Objectives of the chapter are:

1. They are able to explain the computational procedure of LU decomposition.
2. They are able to explain LUP algorithm with examples.

Once the student has completed this topic/ chapter he/she will be able to answer following questions/perform the following activities with Levels of Bloom's Taxonomy:

1. Solve the equation by LUP decomposition. (Level 5)

$$\begin{bmatrix}1 & 5 & 4\\ 2 & 0 & 3\\ 5 & 8 & 2\end{bmatrix}\begin{bmatrix}x_1\\ x_2\\ x_3\end{bmatrix} = \begin{bmatrix}12\\ 9\\ 5\end{bmatrix}$$

2. Write LUP algorithm with example. (Level 3)

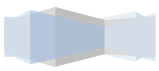
HOME WORK:

1. Solve the equation by LUP decomposition. (like question no. 1)

$$\begin{bmatrix}1 & 5 & 4\\ 2 & 0 & 3\\ 5 & 8 & 2\end{bmatrix}\begin{bmatrix}x_1\\ x_2\\ x_3\end{bmatrix} = \begin{bmatrix}12\\ 9\\ 5\end{bmatrix}$$

2. Solve the equation by LUP decomposition. (like question no. 1)

$$\begin{bmatrix}1 & 0 & 0\\ 4 & 1 & 0\\ -7 & 6 & 1\end{bmatrix}\begin{bmatrix}x_1\\ x_2\\ x_3\end{bmatrix} = \begin{bmatrix}3\\ 14\\ -7\end{bmatrix}$$

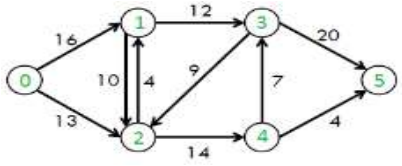


Tutorial: 10

Title: **Network Flow**

Date: **21st and 29th September,2016**, Day: **Wednesday and Thursday**

1) Use the Ford-Fulkerson algorithm to find the maximum flow for the following network. (Source : 0 & Sink: 5) and also find the cuts.



CHAPTER: 13

Title: **Matrix Manipulation Algorithm**

Date: **23rd September, 2016**, Day: **Friday**

CONTENTS

Inversion of Matrix with example.

Chapter Objectives: They are capable to make an algorithm on the basis of matrix manipulation and judge the efficiency of the algorithm.

Broad Objectives of the chapter are:

1. They are able to find out the inverse of a matrix using Gauss-Jordan's rule.

Once the student has completed this topic/ chapter he/she will be able to answer following questions/perform the following activities with Levels of Bloom's Taxonomy:

1. Find out inverse of a matrix using Gauss-Jordan's rule. (Level 5)

$$\begin{bmatrix} 1 & 5 & 4 \\ 2 & 0 & 3 \\ 5 & 8 & 2 \end{bmatrix}$$

HOME WORK:

1. Find out inverse of a matrix using Gauss-Jordan's rule. (like question no. 1)

$$\begin{bmatrix} 1 & 5 & 4 \\ 2 & 0 & 3 \\ 5 & 8 & 2 \end{bmatrix}$$

2. Find out inverse of a matrix using Gauss-Jordan's rule. (like question no. 1)

$$\begin{bmatrix} 1 & 0 & 0 \\ 4 & 1 & 0 \\ -7 & 6 & 1 \end{bmatrix}$$

CHAPTER: 13

Title: **Matrix Manipulation Algorithm**

Date: **26th September, 2016**, Day: **Monday**

CONTENTS

Strassen's Matrix Multiplication algorithm with example and analysis.

Chapter Objectives: They are capable to make an algorithm on the basis of matrix manipulation and judge the efficiency of the algorithm.

Broad Objectives of the chapter are:

1. They are able to explain the computational procedure of Strassen's Matrix Multiplication algorithm with an example.
2. They are able to explain the running time complexity of Strassen's Matrix Multiplication algorithm.

Once the student has completed this topic/ chapter he/she will be able to answer following questions/perform the following activities with Levels of Bloom's Taxonomy:

1. Find C=AB. (Level 5)

$$A=\begin{pmatrix}1 & 2 & 3 & 4\\ 4 & 5 & 6 & 7\\ 2 & 1 & 4 & 1\\ 5 & 1 & 1 & 1\end{pmatrix} \; and \; B = \begin{pmatrix}3 & 1 & 2 & 1\\ 3 & 1 & 1 & 2\\ 4 & 1 & 2 & 2\\ 1 & 1 & 1 & 3\end{pmatrix}$$

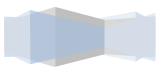
2. Write short notes on Strassen's Matrix Multiplication algorithm. (Level 2)

HOME WORK:

1. Find C=AB. (like question no. 1)

$$A=\begin{pmatrix}1 & 2 & 3 & 4\\ 4 & 5 & 6 & 7\\ 2 & 1 & 4 & 1\\ 5 & 1 & 1 & 1\end{pmatrix} \; and \; B = \begin{pmatrix}3 & 1 & 2 & 1\\ 3 & 1 & 1 & 2\\ 4 & 1 & 2 & 2\\ 1 & 1 & 1 & 3\end{pmatrix}$$

2. Calculate the running time complexity of Strassen's Matrix Multiplication algorithm. (like question no. 2)



CHAPTER: 14

Title: **Notion of NP Completeness**

Date: **28th September, 2016**, Day: **Wednesday**

CONTENTS

Complexity theory : P, NP, NP-hard class, NP-complete class

Chapter Objectives: They are capable to understand the notion of NP Completeness.

Broad Objectives of the chapter are:

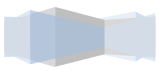
1. They are able to explain P, NP, NP hard, NP Complete class.
2. They are able to explain relation between P, NP, NP hard class, NP Complete class.

Once the student has completed this topic/ chapter he/she will be able to answer following questions/perform the following activities with Levels of Bloom's Taxonomy:

1. Explain P, NP, NP hard, NP Complete class.(Level 4)
2. Write deterministic algorithm (like linear search) with example. (Level 5)
3. Write non deterministic algorithm (like linear search) with example. (Level 5)

HOME WORK:

1. Write non deterministic algorithm on sorting technique with an example. (like question no. 3)
2. Draw a ven diagram on P, NP, NP hard, NP Complete class. (like question no 1)



Tutorial: 11

Title: LUP & Matrix Inversion

Date: **28th and 20th October,2016**, Day: **Wednesday and Thursday**

1) Solve the equation by LUP decomposition.

$$\begin{bmatrix}1 & 5 & 4\\ 2 & 0 & 3\\ 5 & 8 & 2\end{bmatrix}\begin{bmatrix}x_1\\ x_2\\ x_3\end{bmatrix}=\begin{bmatrix}12\\ 9\\ 5\end{bmatrix}$$

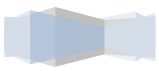
2) Solve the equation by LUP decomposition.

$$\begin{bmatrix}1 & 0 & 0\\ 4 & 1 & 0\\ -7 & 6 & 1\end{bmatrix}\begin{bmatrix}x_1\\ x_2\\ x_3\end{bmatrix}=\begin{bmatrix}3\\ 14\\ -7\end{bmatrix}$$

3) Find out the A^{-1} for the given A.

$$A=\begin{bmatrix}1 & 5 & 4\\ 2 & 0 & 3\\ 5 & 8 & 2\end{bmatrix}$$

4) Show that $AA^{-1} = I$. (relative to Q3)



CHAPTER: 14

Title: **Notion of NP Completeness**

Date: **1st October, 2016**, Day: **Saturday**

CONTENTS

Discuss on optimization problems and Decision problems and relation between them.

Chapter Objectives: They are capable to understand the notion of NP Completeness.

Broad Objectives of the chapter are:

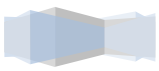
1. They are able to explain verification algorithm.
2. They are able to explain polynomial-time verification algorithm.
3. They are able to explain polynomial time reduction.
4. They are able to clarify optimization versus decision algorithm.

Once the student has completed this topic/ chapter he/she will be able to answer following questions/perform the following activities with Levels of Bloom's Taxonomy:

1. What is polynomial-time verification algorithm? (Level 1)
2. What polynomial time reduction? (Level 1)
3. Convert graph coloring decision problem to optimization problem and vice versa. (Level 4)

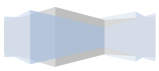
HOME WORK:

1. Convert K-clique decision problem to optimization problem and vice versa.



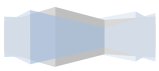
CHAPTER: 14 Title: **Notion of NP Completeness** Date: **3rd October, 2016**, Day: **Monday**
CONTENTS SAT, 3-SAT problems
Chapter Objectives: They are capable to understand the notion of NP Completeness. **Broad Objectives of the chapter are:** 1. They are able to explain Cook's Theorem. 2. They are able to explain SAT problem. 3. They are able to proof 3-SAT is NP complete.
Once the student has completed this topic/ chapter he/she will be able to answer following questions/perform the following activities with Levels of Bloom's Taxonomy: 1. Explain Cook's theorem? (Level 2) 2. Describe SAT problem? (Level 2) 3. Prove that 3-SAT is NP complete. (Level 5)

CHAPTER: 14 Title: **Notion of NP Completeness** Date: **5th October, 2016**, Day: **Wednesday**
CONTENTS Clique decision problem
Chapter Objectives: They are capable to understand the notion of NP Completeness. **Broad Objectives of the chapter are:** 1. They are able to explain Cook's Theorem. 2. They are able to explain K-Clique problem. 3. They are able to proof Clique decision problem is NP complete.
Once the student has completed this topic/ chapter he/she will be able to answer following questions/perform the following activities with Levels of Bloom's Taxonomy: 4. Explain Cook's theorem? (Level 2) 5. Describe K-Clique problem? (Level 2) 6. Prove that Clique decision problem is NP complete. (Level 5)
QUIZ: Not Required.



Title: **Discussion on University QP(Last 5 Years)** Date: **17th & 19th October,2016.** Day**: Monday and Wednesday**
CONTENTS Last 5 years university question paper.
Topic/Unit/Chapter Objectives: we provide discussion on university question paper so that our students can clear their concept and their answers can be to the point. **Broad Objectives of the chapter/topic are:** 1. They are able to explain to analyze, investigate and evaluate. 2. They are able to judge how to apply theory.
Once the student has completed this topic/ chapter he/she will be able to answer following questions/perform the following activities with Levels of Bloom's Taxonomy: **Discussion most of the university questions in last 5 years.**

Title: **Discussion on University QP(Last 5 Years)** Date: **21st & 22nd October,2016.** Day**: Friday and Saturday**
CONTENTS Last 5 years university question paper.
Topic/Unit/Chapter Objectives: we provide discussion on university question paper so that our students can clear their concept and their answers can be to the point. **Broad Objectives of the chapter/topic are:** 1. They are able to explain to analyze, investigate and evaluate. 2. They are able to judge how to apply theory.
Once the student has completed this topic/ chapter he/she will be able to answer following questions/perform the following activities with Levels of Bloom's Taxonomy: **Discussion most of the university questions in last 5 years.**



l) Teaching Strategy/Method (describe instructional methods, usage of ICT, efficient and engaging instructions and display the best practices on institutional website)

1) Taking interactive classes through different examples.
2) Conducting Question – answer session at the end of the class.
3) Real life application for better understanding.

m) Strategy to support weak students

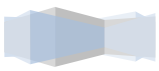
1) To engage the weak students in habit of studying, I give them some easy questions in regular basis.
2) Some weak students also have the problem of forgetting what they have learnt. In my class I always give some tips on how to recall and how to write systematically.
3) Weak students need special attention even after college hours. I always give some extra hours to weak students.

n) Strategy to encourage bright students

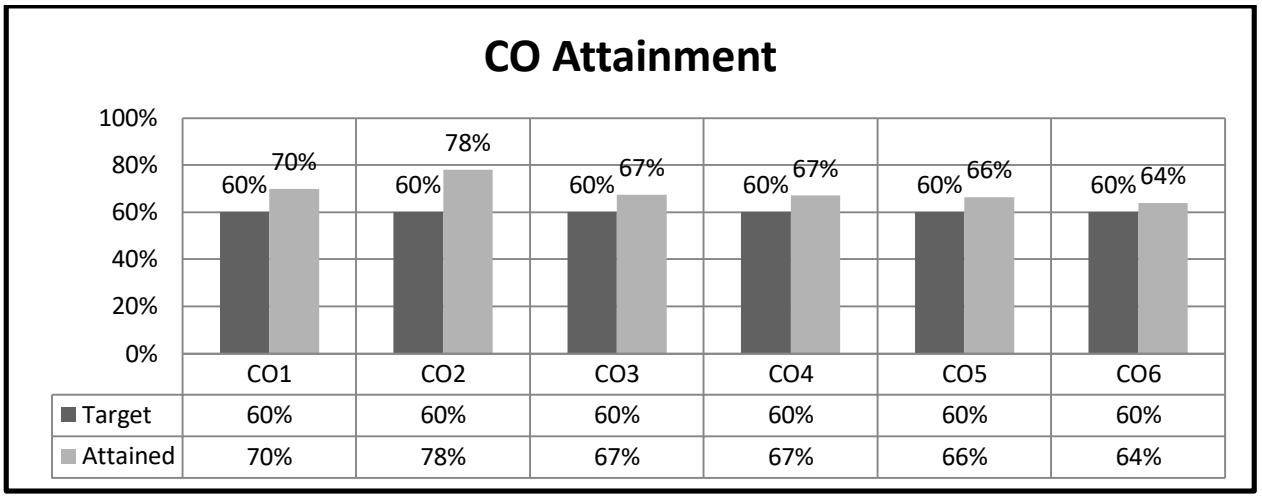
1) Have an extra challenge ready that allows the student to go deeper into the subject, learn a little more, or apply a skill he has just learned in a new way.
2) Some students are engaged with the final year students for their final projects.

o) Efforts to keep students engaged

1) Regular basis Home Work.
2) 5-10 minutes spent in an every class for question answer session.
3) Quiz on regular basis.
4) Some technical assignments are given group wise.



p) Analysis of Students performance in the course (internal) (labs, seminars, tests, assignments, quiz, exam etc)



Comments:

- 70% students have attained the set target of 60% marks for CO1
- 78% students have attained the set target of 60% marks for CO2
- 67% students have attained the set target of 60% marks for CO3
- 67% students have attained the set target of 60% marks for CO4
- 66% students have attained the set target of 60% marks for CO5
- 64% students have attained the set target of 60% marks for CO6

q) Analysis of Students performance in the course (university results)

	Target Course Outcome%	TOTAL STUDENTS	TOTAL STUDENT WHO ATTAINED OUTCOME	% STUDENTS WHO ATTAINED THE OUTCOME
University Result	**60%**	**41**	**41**	**100%**

- 100% students have attained the set target of 60% marks for University Exams.

r) Analysis of Student Feed Back

s) Teacher Self Assessment (at the completion of course)

From the analysis of the results obtained it can be seen that set targets for the course outcome have been achieved successfully by the students. Since this subject is a pre-requisite for Design & Analysis of Algorithm in 5th semester, more emphasis must be given for NP completeness.

t) Recommendations/Suggestions for improvement by faculty

- More emphasis should be given to clear the concepts of NP Completeness.
- Tutorials must be incorporated in the syllabus.
- Increase the total contact hours for theory to 40 hrs, with 4L per week.

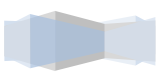
Siliguri Institute of Technology
INTERNAL ASSESSMENT REPORT
Paper Name: Design & Analysis of Algorithm
Paper Code: CS 501

FACULTY NAME : **Mr. MITHUN ROY** **YEAR: 2015**

STREAM: **B.TECH[CSE]** YEAR: **3RD** SEMESTER: **IST** SECTION: **A** NO. OF CLASS HELD: **52**

SN	NAME	ROLL NO.	ATTENDANCE [5 MARKS]		MARKS IN INTERNAL EXAM			ASSIGNMENT [10 MARKS] MARKS=[((I+II +III)/55)*100]/10				TOTAL [30 MARKS]
			TOTAL %	MARKS	I	II	AVG	A-I [15]	A-II [20]	A-III [20]	MARKS	
1	ABHISHEK DEY	11900113001	53.85	3	21	14	9	14	19	14	9	21
2	ADITYA SAHA	11900113002	51.28	3	21	20	10	13	20	14	9	22

3	AKANKSHA KUMARI	11900113003	58.97	3	19	18	9	12	18	13	8	20
4	AKHILESH SINGH	11900113004	43.59	3	20	17	9	13	18	13	8	20
5	AMIT KUMAR	11900113005	79.49	4	22	20	11	15	19	15	9	24
6	AMRITA KUNDU	11900113007	46.15	3	14	21	9	14	17	15	8	20
7	ANGSHUMAN HALDER	11900113008	56.41	3	13	12	6	13	17	16	8	17
8	ANIRBAN DUTTA	11900113009	69.23	3	13	13	7	13	20	14	9	19
9	ANKITA GUPTA	11900113011	76.92	4	22	25	12	14	18	15	9	25
10	ANURAG SHARMA	11900113012	53.85	3	18	16	9	12	19	15	8	20
11	AYUSH AMAN	11900113013	74.36	4	19	13	8	14	20	15	9	21
12	BASANT RAJ	11900113014	64.1	3	13	17	8	15	20	16	9	20
13	BHAWESH PRASAD	11900113016	51.28	3	12	16	7	12	20	15	9	19
14	BINITA AGARWAL	11900113017	46.15	3	14	22	9	14	19	16	9	21
15	BISWAJIT DOLUI	11900113018	66.67	3	10	19	7	13	17	15	8	18
16	CHIRANJIB MUKHERJEE	11900113019	58.97	3	12	14	7	15	17	15	9	19
17	GANESH CHANDRA SAHA	11900113020	69.23	3	14	12	7	13	18	13	8	18
18	JAYDEET KARMAKAR	11900113021	48.72	3	20	10	8	13	18	13	8	19
19	JUHI RANI	11900113022	76.92	4	17	9	7	13	19	14	8	19
20	JYOTI SINHA	11900113023	82.05	5	16	13	7	13	18	14	8	20
21	KARISHMA KUMARI	11900113024	94.87	5	29	23	13	13	19	13	8	26
22	KRITIKA BIBHU	11900113025	89.74	5	28	26	14	12	20	16	9	28
23	KUMAR NISHANT	11900113026	61.54	3	12	3	4	12	19	16	9	16
24	KUNAL KUMAR	11900113027	51.28	3	6	3	2	12	20	15	9	14
25	MILAN SHIT	11900113028	66.67	3	15	10	6	14	18	16	9	18
26	MOHAMMAD MAYAR ALAM	11900113030	64.1	3	22	8	8	12	17	15	8	19
27	MONALISA SINHA	11900113031	61.54	3	17	5	6	15	20	14	9	18
28	MRINAL BARMAN	11900113032	46.15	3	ABS	10	3	12	18	16	8	14
29	NEHA GOYAL	11900113033	56.41	3	22	16	10	15	17	14	8	21
30	NEHA SINGH	11900113034	53.85	3	28	15	11	15	17	14	8	22
31	NIRAJ SONAR	11900113035	64.1	3	16	15	8	12	18	13	8	19
32	PRABHAKAR PAUL	11900113036	64.1	3	21	12	8	13	18	13	8	19
33	PRAGYA KUMARI	11900113037	46.15	3	17	9	7	13	18	14	8	18
34	PRASANJIT BANIK	11900113038	82.05	5	17	16	8	12	20	15	9	22
35	PRITAM KUMAR GHOSH	11900113039	66.67	3	21	15	9	13	19	15	9	21
36	PRITI KUMARI	11900113040	87.18	5	26	16	11	15	17	16	9	25
37	PRIYANKA KUMARI	11900113041	71.79	4	21	14	9	13	17	13	8	21
38	PRONIL CHAKRABORTY	11900113042	58.97	3	21	16	9	13	18	13	8	20



Siliguri Institute of Technology
ATTENDANCE SHEET (LECTURE)
Paper Name: Design & Analysis of Algorithm
Paper Code: CS 501

ROLL NO.	20/7	21/7	23/7	27/7	28/7	3/8	4/8	6/8	10/8	11/8	11/9	13/8	17/8	24/8	25/8	25/8	27/8	27/8	31/8	1/9	1/9	3/9	7/9	8/9	8/9	10/9	14/9	15/9	15/9	28/9	29/9	1/10	5/10	6/10	3/11	3/11	5/11	5/11
11900113001	0	0	0	0	0	0	1	1	1	1	1	1	1	0	1	1	0	0	1	1	1	0	0	0	0	1	1	1	1	1	0	1	1	1	0	0	0	0
11900113002	0	0	0	0	0	0	0	0	1	1	1	0	1	1	1	1	1	1	1	0	0	0	0	1	1	1	1	0	0	0	0	1	1	1	0	0	1	1
11900113003	0	0	0	0	0	1	0	1	1	1	1	1	1	0	1	1	1	1	0	1	1	1	0	1	1	1	0	1	1	1	0	1	1	0	0	0	0	0
11900113004	1	0	0	0	0	1	1	1	1	1	1	1	1	0	0	0	0	0	1	0	0	0	1	0	0	1	0	1	1	1	0	1	0	0	0	0	0	0
11900113005	1	0	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	0	1	1	0	1	1	1	1	1	1	1	1	0	1	1	0	0	0	1	1

Siliguri Institute of Technology
ATTENDANCE SHEET (TUTORIAL CLASS)
Paper Name: Design & Analysis of Algorithm
Paper Code: CS 501

FACULTY NAME : **Mr. MITHUN ROY** **YEAR: 2015**

STREAM: **B.TECH[CSE]** YEAR: **3RD** SEMESTER: **IST** SECTION: **A** NO. OF CLASS HELD: **11**

SN	NAME	ROLL NO.	24/7	31/7	7/8	14/8	21/8	28/8	4/9	11/9	21/9	9/10	2/11	Total
1	ABHISHEK DEY	11900113001	1	1	1	0	1	0	1	1	1	0	0	7
2	ADITYA SAHA	11900113002	1	1	0	1	1	0	0	1	1	0	0	6
3	AKANKSHA KUMARI	11900113003	1	0	0	1	0	0	0	1	1	0	0	4
4	AKHILESH SINGH	11900113004	1	1	0	1	0	0	1	1	0	0	1	6
5	AMIT KUMAR	11900113005	1	0	1	0	1	0	1	0	1	1	1	7
6	AMRITA KUNDU	11900113007	0	0	0	0	0	1	0	0	0	0	0	1
7	ANGSHUMAN HALDER	11900113008	1	1	1	0	1	0	1	0	1	0	0	6
8	ANIRBAN DUTTA	11900113009	0	0	1	0	0	0	0	0	1	0	1	3
9	ANKITA GUPTA	11900113011	0	0	0	0	0	1	1	1	1	1	1	6
10	ANURAG SHARMA	11900113012	1	0	1	0	0	1	0	1	1	0	1	6
11	AYUSH AMAN	11900113013	0	1	1	0	1	0	1	1	1	1	1	8
12	BASANT RAJ	11900113014	0	1	1	1	0	1	0	0	1	1	0	6
13	BHAWESH PRASAD	11900113016	0	1	0	0	1	0	1	1	0	0	1	5
14	BINITA AGARWAL	11900113017	0	1	1	0	1	1	0	0	0	1	1	6
15	BISWAJIT DOLUI	11900113018	0	1	0	0	1	0	1	1	0	0	0	4
16	CHIRANJIB MUKHERJEE	11900113019	1	0	0	1	1	0	0	0	0	0	1	4
17	GANESH CHANDRA SAHA	11900113020	1	1	0	1	1	0	0	1	0	1	0	6
18	JAYDEET KARMAKAR	11900113021	0	0	1	0	1	0	0	1	1	0	0	4
19	JUHI RANI	11900113022	1	1	0	0	0	1	0	0	1	1	0	5
20	JYOTI SINHA	11900113023	1	0	0	1	1	0	0	0	1	0	1	5
21	KARISHMA KUMARI	11900113024	1	0	1	0	0	0	0	0	0	0	0	2
22	KRITIKA BIBHU	11900113025	1	1	0	0	1	1	1	1	1	0	0	7
23	KUMAR NISHANT	11900113026	1	0	1	1	0	0	0	1	1	1	0	6
24	KUNAL KUMAR	11900113027	0	1	1	0	0	0	1	0	0	0	1	4
25	MILAN SHIT	11900113028	1	1	1	1	1	1	1	0	0	1	0	8
26	MOHAMMAD MAYAR ALAM	11900113030	1	0	1	1	1	1	1	1	0	0	0	7
27	MONALISA SINHA	11900113031	0	1	1	0	0	1	1	0	1	0	0	5
28	MRINAL BARMAN	11900113032	1	0	1	1	1	0	0	0	0	1	0	5
29	NEHA GOYAL	11900113033	0	1	0	0	0	0	0	1	1	0	0	3

30	NEHA SINGH	11900113034	1	0	1	0	1	0	1	0	1	0	0	5
31	NIRAJ SONAR	11900113035	1	0	1	1	1	1	0	0	0	0	1	6
32	PRABHAKAR PAUL	11900113036	0	0	1	1	1	0	1	1	1	0	0	6
33	PRAGYA KUMARI	11900113037	0	0	1	1	1	1	0	0	1	1	0	6
34	PRASANJIT BANIK	11900113038	1	1	1	0	1	0	1	1	0	1	1	8
35	PRITAM KUMAR GHOSH	11900113039	0	0	0	0	1	1	0	0	1	0	0	3
36	PRITI KUMARI	11900113040	0	0	0	0	1	0	1	1	1	1	1	6
37	PRIYANKA KUMARI	11900113041	0	0	1	1	1	0	0	0	1	0	1	5
38	PRONIL CHAKRABORTY	11900113042	0	0	1	0	0	0	1	1	0	0	0	3

Siliguri Institute of Technology
LABORATORY ATTENDANCE SHEET
Paper Name: Design & Analysis of Algorithm
Paper Code: CS 591

FACULTY NAME : **Mr. MITHUN ROY** **YEAR: 2015**

STREAM: **B.TECH[CSE]** YEAR: **3RD** SEMESTER: **IST** GROUP: **A1** NO. OF PRACTICAL HELD: **10**

SN	NAME	DAY	1	2	3	4	5	6	7	8	9	10	TOTAL MARKS
		DATE	30/7	6/8	13/8	20/8	27/8	3/9	10/9	17/9	1/10	5/11	
		ROLL NO											
1	ABHISHEK DEY	11900113001	0	1	1	1	0	1	1	1	1	0	**7**
2	ADITYA SAHA	11900113002	1	1	1	1	0	0	1	0	1	0	**6**
3	AKANKSHA KUMARI	11900113003	0	1	1	1	1	1	1	0	1	0	**7**
4	AKHILESH SINGH	11900113004	0	1	1	1	0	0	1	1	0	0	**5**
5	AMIT KUMAR	11900113005	0	1	1	1	1	1	1	1	1	0	**8**
6	AMRITA KUNDU	11900113007	1	1	1	1	1	0	1	0	1	0	**7**
7	ANGSHUMAN HALDER	11900113008	1	1	1	1	1	1	0	0	1	0	**7**
8	ANIRBAN DUTTA	11900113009	1	1	0	1	1	1	1	1	1	0	**8**
9	ANKITA GUPTA	11900113011	1	1	1	1	1	1	1	1	1	1	**10**
10	ANURAG SHARMA	11900113012	1	1	1	1	1	1	0	1	1	0	**8**
11	AYUSH AMAN	11900113013	0	1	1	1	0	0	1	1	1	0	**6**
12	BASANT RAJ	11900113014	0	1	1	0	1	0	1	1	1	1	**7**
13	BHAWESH PRASAD	11900113016	1	1	1	1	1	1	1	0	1	0	**8**
14	BINITA AGARWAL	11900113017	1	1	1	1	1	1	1	1	1	0	**9**
15	BISWAJIT DOLUI	11900113018	1	1	1	1	1	1	1	1	1	0	**9**
16	CHIRANJIB MUKHERJEE	11900113019	1	1	1	1	0	0	0	1	1	0	**6**

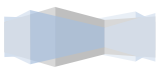
17	GANESH CHANDRA SAHA	11900113020	1	1	1	1	1	1	1	1	1	0	**9**
18	JAYDEET KARMAKAR	11900113021	1	1	1	1	1	0	1	0	1	0	**7**
19	JUHI RANI	11900113022	1	1	1	1	1	1	1	1	1	0	**9**

Siliguri Institute of Technology
LABORATORY ATTENDANCE SHEET
Paper Name: Design & Analysis of Algorithm
Paper Code: CS 591

FACULTY NAME : **Mr. MITHUN ROY** **YEAR: 2015**

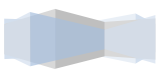
STREAM: **B.TECH[CSE]** YEAR: **3RD** SEMESTER: **IST** GROUP: **A2** NO. OF PRACTICAL HELD: **10**

SN	NAME	DAY	1	2	3	4	5	6	7	8	9	10	TOTAL MARKS
		DATE	**29/7**	**5/8**	**12/8**	**19/8**	**26/8**	**9/9**	**16/9**	**30/9**	**7/10**	**4/11**	
		ROLL NO											
1	JYOTI SINHA	11900113023	1	1	1	1	1	1	1	1	1	0	**9**
2	KARISHMA KUMARI	11900113024	0	1	1	1	1	1	1	1	1	0	**8**
3	KRITIKA BIBHU	11900113025	1	1	1	1	1	1	1	1	1	1	**10**
4	KUMAR NISHANT	11900113026	1	0	1	1	1	1	1	0	1	1	**8**
5	KUNAL KUMAR	11900113027	1	0	0	0	1	1	1	0	0	0	**4**
6	MILAN SHIT	11900113028	1	0	1	1	0	1	1	1	1	1	**8**
7	MOHAMMAD MAYAR ALAM	11900113030	1	1	0	1	0	1	1	0	1	1	**7**
8	MONALISA SINHA	11900113031	1	0	1	1	1	1	0	1	1	0	**7**
9	MRINAL BARMAN	11900113032	1	1	0	1	0	1	1	1	1	0	**7**
10	NEHA GOYAL	11900113033	1	1	1	1	1	1	1	1	0	0	**8**
11	NEHA SINGH	11900113034	1	0	1	1	1	0	1	1	1	1	**8**
12	NIRAJ SONAR	11900113035	1	0	0	1	1	0	1	1	0	0	**5**
13	PRABHAKAR PAUL	11900113036	1	1	1	1	1	1	1	1	0	1	**9**
14	PRAGYA KUMARI	11900113037	1	1	0	1	1	1	0	1	0	0	**6**
15	PRASANJIT BANIK	11900113038	1	0	1	0	1	1	0	1	1	0	**6**
16	PRITAM KUMAR GHOSH	11900113039	1	0	1	0	1	1	1	0	1	0	**6**
17	PRITI KUMARI	11900113040	1	1	1	1	1	1	1	1	1	0	**9**
18	PRIYANKA KUMARI	11900113041	0	1	1	1	1	1	1	1	0	0	**7**
19	PRONIL CHAKRABORTY	11900113042	1	0	1	0	1	1	1	1	0	0	**6**



Siliguri Institute of Technology
RECORDS OF ASSIGNMENTS/~~QUIZ~~
Paper Name: Design & Analysis of Algorithm
Paper Code: CS 591

SN	NAME	ROLL NO.	Assign - I	Assign - II	Assign - III	SN	NAME	ROLL NO.	Assign - I	Assign - II	Assign - III
1	ABHISHEK DEY	11900113001	1	1	1	20	JYOTI SINHA	11900113023	1	1	1
2	ADITYA SAHA	11900113002	1	1	1	21	KARISHMA KUMARI	11900113024	1	1	1
3	AKANKSHA KUMARI	11900113003	1	1	1	22	KRITIKA BIBHU	11900113025	1	1	1
4	AKHILESH SINGH	11900113004	1	1	1	23	KUMAR NISHANT	11900113026	1	1	1
5	AMIT KUMAR	11900113005	1	1	1	24	KUNAL KUMAR	11900113027	1	1	1
6	AMRITA KUNDU	11900113007	1	1	1	25	MILAN SHIT	11900113028	1	1	1
7	ANGSHUMAN HALDER	11900113008	1	1	1	26	MOHAMMAD MAYAR ALAM	11900113030	1	1	1
8	ANIRBAN DUTTA	11900113009	1	1	1	27	MONALISA SINHA	11900113031	1	1	1
9	ANKITA GUPTA	11900113011	1	1	1	28	MRINAL BARMAN	11900113032	1	1	1
10	ANURAG SHARMA	11900113012	1	1	1	29	NEHA GOYAL	11900113033	1	1	1
11	AYUSH AMAN	11900113013	1	1	1	30	NEHA SINGH	11900113034	1	1	1
12	BASANT RAJ	11900113014	1	1	1	31	NIRAJ SONAR	11900113035	1	1	1
13	BHAWESH PRASAD	11900113016	1	1	1	32	PRABHAKAR PAUL	11900113036	1	1	1
14	BINITA AGARWAL	11900113017	1	1	1	33	PRAGYA KUMARI	11900113037	1	1	1
15	BISWAJIT DOLUI	11900113018	1	1	1	34	PRASANJIT BANIK	11900113038	1	1	1
16	CHIRANJIB MUKHERJEE	11900113019	1	1	1	35	PRITAM KUMAR GHOSH	11900113039	1	1	1
17	GANESH CHANDRA SAHA	11900113020	1	1	1	36	PRITI KUMARI	11900113040	1	1	1
18	JAYDEET KARMAKAR	11900113021	1	1	1	37	PRIYANKA KUMARI	11900113041	1	1	1
19	JUHI RANI	11900113022	1	1	1	38	PRONIL CHAKRABORTY	11900113042	1	1	1



Siliguri Institute of Technology
LIST OF PRACTICAL'S
Paper Name: Design & Analysis of Algorithm Lab
Paper Code: CS 591

SN	Details of Experiment(s)	Hours Allotted
1	Experiment on different **Searching** Techniques and also judge the running time complexity. **1) Linear Search** **2) Binary Search**	3 HRS
2	Experiment on different **Sorting** techniques and also judge the running time complexity. **3) Merge Sort** **4) Quick Sort**	3 HRS
3	Experiment on different **Sorting** techniques and also judge the running time complexity. **5) Heap Sort** **6) Counting Sort**	3 HRS
4	Experiment on some **recursion** problems also judge the running time complexity as well as plot the graph. **7) Calculate x^y** **8) N^{th} Fibonacci Number** **9) Tower of Hanoi etc and**	3 HRS
5	Experiment on **Dynamic Programming** algorithm strategy and also judge the running time complexity. **10) Matrix Chain Multiplication.**	3 HRS
6	Experiment on **Dynamic Programming** algorithm strategy and also judge the running time complexity. **11) Floyd's Algorithm**	3 HRS
7	Experiment on **Backtracking** algorithm strategy and also judge the running time complexity. **12) 4 Queen** **13) Graph Coloring**	3 HRS

8	Experiment on **Minimum Spanning Tree** and also judge the running time complexity. (Any one) **14) Prim's Algorithm** **15) Kruskal's Algorithm**	3 HRS
9	Experiment on **Graph Traversal Techniques** and also judge the running time complexity. **16) BFS** **17) DFS**	3 HRS
10	Experiment on **String Matching** Algorithm and also judge the running time complexity. **18) KMP**	3 HRS

Siliguri Institute of Technology
SESSIONAL/PRACTICAL PERFORMANCE RECORD
Paper Name: Design & Analysis of Algorithm Lab
Paper Code: CS 591

FACULTY NAME : **Mr. MITHUN ROY** **YEAR: 2015**

STREAM: **B.TECH[CSE]** YEAR: **3RD** SEMESTER: **IST** SECTION: **A**

SN	NAME	ROLL NO	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	TOTAL[40]
1	ABHISHEK DEY	11900113001	0	4	4	4	0	4	4	4	4	0	**28**
2	ADITYA SAHA	11900113002	4	4	4	4	0	0	4	0	4	0	**24**
3	AKANKSHA KUMARI	11900113003	0	4	4	4	4	4	4	0	4	0	**28**
4	AKHILESH SINGH	11900113004	0	4	4	4	0	0	4	4	0	0	**20**
5	AMIT KUMAR	11900113005	0	4	4	4	4	4	4	4	4	0	**32**
6	AMRITA KUNDU	11900113007	4	4	4	4	4	0	4	0	4	0	**28**
7	ANGSHUMAN HALDER	11900113008	4	4	4	4	4	4	0	0	4	0	**28**
8	ANIRBAN DUTTA	11900113009	4	4	0	4	4	4	4	4	4	0	**32**
9	ANKITA GUPTA	11900113011	4	4	4	4	4	4	4	4	4	4	**40**
10	ANURAG SHARMA	11900113012	4	4	4	4	4	4	0	4	4	0	**32**
11	AYUSH AMAN	11900113013	0	4	4	4	0	0	4	4	4	0	**24**
12	BASANT RAJ	11900113014	0	4	4	0	4	0	4	4	4	4	**28**
13	BHAWESH PRASAD	11900113016	4	4	4	4	4	4	4	0	4	0	**32**
14	BINITA AGARWAL	11900113017	4	4	4	4	4	4	4	4	4	0	**36**
15	BISWAJIT DOLUI	11900113018	4	4	4	4	4	4	4	4	4	0	**36**
16	CHIRANJIB MUKHERJEE	11900113019	4	4	4	4	0	0	0	4	4	0	**24**
17	GANESH CHANDRA SAHA	11900113020	4	4	4	4	4	4	4	4	4	0	**36**
18	JAYDEET KARMAKAR	11900113021	4	4	4	4	4	0	4	0	4	0	**28**

19	JUHI RANI	11900113022	4	4	4	4	4	4	4	4	4	0	**36**
20	JYOTI SINHA	11900113023	4	4	4	4	4	4	4	4	4	0	**36**
21	KARISHMA KUMARI	11900113024	0	4	4	4	4	4	4	4	4	0	**32**
22	KRITIKA BIBHU	11900113025	4	4	4	4	4	4	4	4	4	4	**40**
23	KUMAR NISHANT	11900113026	4	0	4	4	4	4	4	0	4	4	**32**
24	KUNAL KUMAR	11900113027	4	0	0	0	4	4	4	0	0	0	**16**
25	MILAN SHIT	11900113028	4	0	4	4	0	4	4	4	4	4	**32**
26	MOHAMMAD MAYAR ALAM	11900113030	4	4	0	4	0	4	4	0	4	4	**28**
27	MONALISA SINHA	11900113031	4	0	4	4	4	4	0	4	4	0	**28**
28	MRINAL BARMAN	11900113032	4	4	0	4	0	4	4	4	4	0	**28**
29	NEHA GOYAL	11900113033	4	4	4	4	4	4	4	4	0	0	**32**
30	NEHA SINGH	11900113034	4	0	4	4	4	0	4	4	4	4	**32**
31	NIRAJ SONAR	11900113035	4	0	0	4	4	0	4	4	0	0	**20**
32	PRABHAKAR PAUL	11900113036	4	4	4	4	4	4	4	4	0	4	**36**
33	PRAGYA KUMARI	11900113037	4	4	0	4	4	4	0	4	0	0	**24**
34	PRASANJIT BANIK	11900113038	4	0	4	0	4	4	0	4	4	0	**24**
35	PRITAM KUMAR GHOSH	11900113039	4	0	4	0	4	4	4	0	4	0	**24**
36	PRITI KUMARI	11900113040	4	4	4	4	4	4	4	4	4	0	**36**
37	PRIYANKA KUMARI	11900113041	0	4	4	4	4	4	4	4	0	0	**28**
38	PRONIL CHAKRABORTY	11900113042	4	0	4	0	4	4	4	4	0	0	**24**

NAME WITH ROLL NUMBERS OF STUDENT WHOSE ACADEMIC PERFORMANCE IS NOT SATISFACTORY			
Sl.	**Name of Student**	**Roll No.**	**Remedial measures taken by teacher**
1	ABHISHEK DEY	11900113001	• Additional doubt clearing sessions • Providing extra assignments to students with poor attendance. • Guiding them through previous question papers • Highlighting important and frequently asked questions
2	GANESH CHANDRA SAHA	11900113020	
3	KUNAL KUMAR	11900113027	
4	MILAN SHIT	11900113028	
5	MRINAL BARMAN	11900113032	
6	NIRAJ SONAR	11900113035	

CERTIFICATE

I, the undersigned, have completed the course allotted to me as shown below

Sl. No.	Semester	Subject with Code	Total Chapters	Remarks
1.	5th	Design & Analysis of Algorithm (CS501) Design & Analysis Laboratory (CS 591)	15	

Date :	**Signature of Faculty**

Submitted to HOD

Certificate by HOD

I, the undersigned, certify that **Prof. Mithun Roy** has completed the course work allotted to him satisfactorily / not satisfactorily.

Date :	**Signature of HOD**

Submitted to Director

Date :	**Signature of Director**

Siliguri Institute of Technology
Department of Computer Sc. & Engineering
1st Internal Exam-2015
Design & Analysis of Algorithm, CS 501
F.M-30, Time: 1h 30m

Group- A, Marks allotted: 10

Answer the following questions (2X5 = 10)

1. a) Write the **Master Theorem**.
   b) **Solve** the following recurrence **using Master Theorem**. [CO3]
   $T(n) = 4T\left(\frac{n}{3}\right) + n^2 \lg(n)$

   OR

   a) Prove that $\sum_{k=0}^{n} \log(k)$ is $O(n \lg n)$. [CO3]
2. **Create a Max Heap** for the following key elements. A = {5,3,17,10,84,19,6,22,9} [CO5]

   OR

Perform the **partition** operation once (one time) on the following array as per the requirement of the **quicksort algorithm**, assuming the last element is the pivot of the array. Clearly mention the steps.
$A[] = \{2,8,7,1,3,5,6,4\}$. [CO5]

Group- B, Marks allotted: 10

Answer the following questions (2X5 = 10)

3. Find the recurrence relation of **Binary search** and derive the **time complexity** of **Binary search**. [CO1]

   OR

   Derive the Time complexity of **Mergesort algorithm**. [CO1]

4. Analyze the **Best** case and **Worst** case time complexity of **Quicksort algorithm**. [CO4]

Group- C, Marks allotted: 10

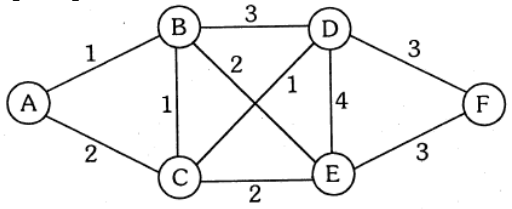
Answer the following questions (2X5 = 10)

5. Write difference between **Divide & Conquer strategy** and **Dynamic Programming** with an example. [CO6]

   OR

   Write short notes on **all pair shortest path** OR **Asymptotic Notation.** [CO6]

6. Find out the **shortest path** between Vertex 'A' to Vertex 'F' using **Dijkstr's algorithm** where Vertex 'A' is the start Vertex. [CO6]



OR

Find an **optimal parenthesization** of a **matrix-chain product** whose sequence of dimensions is $\langle 5,10,3,12,5 \rangle$. [CO6]

Siliguri Institute of Technology
Department of Computer Sc. & Engineering
2nd Internal Exam-2015
Design & Analysis of Algorithm, CS 501
F.M-30, Time: 1h 30m

Group- A, Marks allotted: 10

Answer any two questions (2X5 = 10)

3. Find an **optimal solution** to the **knapsack** instance n=7, m=15, $(v_1, v_2, v_3, \ldots, v_7) = (10, 5, 15, 7, 6, 18, 3)$, and $(w_1, w_2, w_3, \ldots, w_7) = (2, 3, 5, 7, 1, 4, 1)$. **[CO 6]**
4. Design an algorithm that calculate the sum of n elements of an array and also calculate the

running time complexity. **[CO 1]**

OR

5. Compute the prefix function Π for the pattern ababbabbabbababbabb where the alphabet is Σ = {a,b}. **[CO 1]**

Group- B. Marks allotted: 10

Answer any two questions (2X5 = 10)

6. Solve the equation following linear equations using **LUP decomposition**. **[CO 5]**

$$x + 5y + 4z = 12$$
$$2x + 3z = 9$$
$$5x + 8y + 2z = 5$$

7. Discuss the procedure for **Strassen's matrix multiplication** to evaluate the product of 'n' matrices. **[CO 4]**

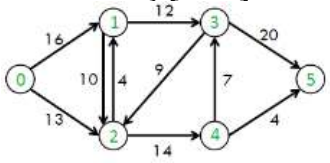
OR

8. Find the resulting recurrence relation for the same (**Strassen's matrix multiplication**) and analyze its time complexity. **[CO 4]**

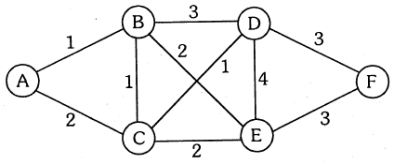
Group- C, Marks allotted: 10

Answer any two questions (2X5 = 10)

9. Write **Max-flow Min-cut** Theorem. Use the Ford-Fulkerson algorithm to find the maximum flow for the following network. (Source : 0 & Sink: 5) **[CO 5]**

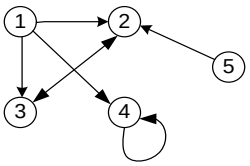


10. Find out the **Minimum Spanning Tree** for the following graph using **Prim's** or **Kruskal's** Algorithm. **[CO 6]**



OR

11. Find out the **DFS tree** and **classified the edges** for the following Graph where start vertex is '1'. **[CO 6]**



Year: 3RD Semester: 1ST Section: A
Marks: 15

Paper Name: Design & Analysis of Algorithm
Paper Code: CS 501
Submission Date: 6th August, 2015
Assignment policy:

- Assignments must be submitted **in class** as hardcopy (A4 sheet) on the due date mentioned in the assignment
- Early submissions are allowed.
- All assignments must be done individually. Anyone cheating will receive a zero for that assignment.
- Late submission policy: **No late submissions will be allowed** on any assignment. However, earlier submissions are allowed at any time before due.

Assignment – I

I. Given 2 arrays A and B, each of size N, 3 + 1 + 2 = 6

1) Design an algorithm to test whether there is at least one common element between the 2 arrays,
2) Prove its correctness, and
3) Estimate its speed.

II. Solve the following recurrence by successive substitutions. F (1) = 1, and 2

$$F(N) = 2F(N-1) + N, \text{ for any } N > 1.$$

III. Which of the following equalities are true and why? 7 X 1 = 7

1) $3N^2 + 6N = O(N^2)$
2) $3N^2 + 6N = O(N^2 \log N)$
3) $N^2 \log N = O(N^2)$
4) $3^N = O(2^N)$
5) $\log N = O((\log \log N)^4)$
6) $N = O((\log N)^{\log N})$

7) $N^{100} = O(2^N)$

Siliguri Institute of Technology

Computer Science & Engineering Department

Year: 3RD Semester: 1ST Section: A

Marks: 20

Paper Name: Design & Analysis of Algorithm

Paper Code: CS 501

Submission Date: 18th August, 2015

Assignment policy:

- Assignments must be submitted **in class** as hardcopy (A4 sheet) on the due date mentioned in the assignment
- Early submissions are allowed.
- All assignments must be done individually. Anyone cheating will receive a zero for that assignment.
- Late submission policy: **No late submissions will be allowed** on any assignment. However, earlier submissions are allowed at any time before due.

Assignment – II

(4 X 5 = 20)

I. By applying the master theorem solve the following recurrences. For the base cases, assume that $T(1) = O(1)$.

1) $T(N) = 25T(N/5) + N^{2.1}$

2) $T(N) = 25T(N/5) + N^{1.5}$

3) $T(N) = 25T(N/5) + N^2$

II. Solve the following recurrence by successive substitutions or by induction. For the base cases, assume that $T(1) = O(1)$.

$T(N) \leq 25T(N/5) + N^2 \log N.$

III. The ELEMENT DISTINCTNESS problem consists of testing whether a given set of N numbers
have no duplicates. Design an O(N log N) step comparison-based algorithm for this problem.

IV. In the selection problem (finding the K^{TH} smallest element), if we group the N elements into N/3 groups each of 3 elements and make appropriate changes to the algorithm, derive the speed of the resulting algorithm. Repeat it when each group consists of 7 elements.

Siliguri Institute of Technology

Computer Science & Engineering Department

Year: 3^{RD} Semester: 1^{ST} Section: A

Marks: 20

Paper Name: Design & Analysis of Algorithm

Paper Code: CS 501

Submission Date: 25th November, 2015

Assignment policy:

- Assignments must be submitted **in class** as hardcopy (A4 sheet) on the due date mentioned in the assignment
- Early submissions are allowed.
- All assignments must be done individually. Anyone cheating will receive a zero for that assignment.
- Late submission policy: **No late submissions will be allowed** on any assignment. However, earlier submissions are allowed at any time before due.

Assignment – III

(4 X 5 = 20)

I. Design a dynamic programming algorithm for the following problem:

Given a sequence $A_1 \star_1 A_2 \star_2 \cdots A_{N-1} \star_{N-1} A_N$, in which each A_I is a positive integer and each $\star_I$ is '+' or '-', compute a parenthesization of the expression such that the resulting value is the maximum possible. It suffices to compute the resulting value instead of the parenthesization. Estimate its speed.

For example, if the given sequence is 3 - 4 -5, ((3-4)-5) results in -6 while (3-(4-5)) results in 4. The second parenthesization results in the maximum possible value, and the output is 4.

II. Let A be an M × N array of numbers. In phase 1, we sort the rows and then in phase 2, we sort the columns. Prove that after both the phases are completed, the rows remain in sorted order.

III. Let $S_1, S_2, \cdots, S_M$ be nonempty subsets of $\{1, 2, \cdots, N\}$, and let the total number of elements in all the S_I S be N. Design an O(N) step algorithm for sorting the S_I S.

IV. In the standard Heap Sort, given any N numbers $A_1, A_2, \cdots, A_N$, we first arranged them into a complete binary tree and then satisfied the heap property in the order $A_N, A_{N-1}, \cdots, A_1$. If we

want to heapify in the order $A_1, A_2, \cdots, A_N$, describe an appropriate algorithm and estimate its speed. (In this algorithm, for any $I \geq 1$, after a heap is constructed for $A_1, A_2, \cdots, A_I$, the next number A_{I+1} is brought into the heap in a bottom-up fashion.)