PRESIDENCY UNIVERSITY

(Established under the Presidency University Act, 2013 of the Karnataka Act 41 of 2013)

[2022-23 EVEN/ WINTER SEMESTER]

COURSE HAND OUT

SCHOOL: SCSE&IS DEPT: COMPUTER SCIENCE AND ENGINEERING DATE OF ISSUE:29-01-2023

NAME OF THE PROGRAM	: B.TECH COMPUTER SCIENCE AND ENGINEERING
P.R.C. APPROVAL REF	: PU/AC18.8/CSE16/CSE/2021-25
SEMESTER/YEAR	:IV / II YEAR
COURSE TITLE & CODE	: DESIGN AND ANALYSIS OF ALGORITHMS/ CSE2007
COURSE CREDIT STRUCTURE	: L-3: P-0: C-3
CONTACT HOURS	: 3 SESSIONS PER WEEK
COURSE IC	: DR. YASHASWINI K A, MR. SUKRUTH GOWDA M A/ SMITHA PATIL
COURSE INSTRUCTOR	: Dr.Murali Parameswaran, Dr.Manjula H.M, Mr.Mohammed Mujeer Ulla, Mr.Mrutyunjaya M.S, Mr.Deepak Raj S, Dr.Yashaswini K.A, Ms.Shruthi U , Ms.Soumya, Ms.Swathi Pai M, Mr.Afroj Alam, Dr.Shankar Ramamoorthy, Mr.Sukruth Gowda M.A, Smitha Patil
COURSE URL	: www.camu.in

PROGRAM OUTCOMES:

PO-1: Engineering knowledge: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

PO-2: Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO-3: Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

PO-4: Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO-5: Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations

PO-6: The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

PO-7: Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of and need for sustainable development.

PO-8: Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO-9: Individual and teamwork: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

PO-10: Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO-11: Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO-12: Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

COURSE PREREQUISITES:

CSE2001- Data Structure and Algorithms, Concept of Problem Solving

COURSE DESCRIPTION:

This intermediate course enables students to design and analyze efficient algorithms to solve problems. This course covers typical design methods such as divide-and-conquer, dynamic programming and greedy method to solve problems. The students shall develop strong analytical skills as part of this course.

Type of Skill: Employability and Skill Development

Nature of the Course: Conceptual and Application Based.

COURSE OUTCOMES:

On successful completion of the course the students shall be able to:

TABLE 1: COURSE OUTCOMES		
CO Number	**CO**	**Expected BLOOMS LEVEL**
CO1	Identify the efficiency of a given algorithm	**Comprehension**
CO2	Employ divide and conquer approach to solve a problem	**Application**
CO3	Illustrate dynamic programming approach to solve a given problem	**Application**
CO4	Solve problem using the greedy method	**Application**
CO5	Discuss the techniques to solve a real-world problem based on its complexity classes	**Comprehension**

MAPPING OF C.O. WITH P.O. : [H-HIGH , M- MODERATE, L-LOW]

TABLE 2: CO PO Mapping ARTICULATION MATRIX												
CO. No	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PO 12
CO1	**M**	**M**	-	-	-	-	-	-	-	-	-	L
CO2	**M**	**H**	**H**	L	L	-	-	-	-	M	-	M
CO3	**M**	**H**	**H**	L	L	-	-	-	-	M	-	M
CO4	**M**	**H**	**H**	L	L	-	-	-	-	M	-	M
CO5	**M**	**H**	**H**	**M**	L	-	-	-	-	M	-	H

COURSE CONTENT

Module 1: Introduction to Algorithms	**[06 sessions] [Comprehension]**
Algorithm Design and efficiency, measuring of running time of algorithms. Asymptotic Growth and Notations. Recurrences--Masters method.	
Module 2: Review of Searching and Sorting techniques	**[11 sessions] [Application]**
Sorting: Insertion sort and merge sort ,Quicksort, Heapsort, Lower bound of comparison-based sorting, non-comparison-based sorting: Radix sort. Search: Review of Linear Search and Binary Search, Hashing and hash tables. Divide and Conquer: Examples. Strassen's Matrix multiplication.	
Module 3: Greedy Algorithms	**[08 sessions] [Application]**
Introduction, Fractional Knapsack Problem, Minimal Spanning Tree: Prim's Algorithm and Kruskal's Algorithm, Single-source Shortest Path: Dijkstra's Algorithm and Huffman Codes.	
Module 4: Dynamic Programming	**[08 sessions] [Application]**
Introduction with examples, Principles of Memorization, 0-1 Knapsack Problem, Bellman-Ford algorithm, Floyd-Warshall's Algorithms. Optimal Binary Search Trees, Chain Matrix Multiplication.	
Module 5: Complexity Classes and Heuristics	**[07 sessions] [Comprehension]**
Complexity classes: P, NP, and NP-Complete Problems. Backtracking: n-Queens. Branch and bound: Travelling Salesman Problem.	

DELIVERY PROCEDURE (PEDAGOGY):

TABLE 3: SPECIAL DELIVERY METHOD/ PEDAGOGY PLANNED WITH TOPICS

S. No	Lecture Number	Subtopic as per lesson Plan	Pedagogy title/ short explanation of adopted pedagogy	Activity Completion Details
1	L6	Recurrences--Masters method	Self-learning : Seminar on problems	
2	L11	Review of Linear Search and Binary Search	Experiential Learning: Flipped Class	
3	L23	Problems on Prims and Kruskals Algorithm	Experiential Learning, Implementation of programs	
4	L31	Floyd-Warshall's Algorithms	Participative Learning: Real scenarios to be solved in group	

REFERENCE MATERIALS:

Text Books:

T1. Thomas H.Cormen, Charles E.Leiserson, Ronald L. Rivest and Clifford Stein, 'Introduction to Algorithms', MIT Press, 2022.

T2. J. Kleinberg and E. Tardos, 'Algorithm Design', Addison-Wesley, 2005.

Reference Books:

R1. Anany Levitin, 'Introduction to the Design and Analysis of Algorithms', Pearson Education, 2003.

R2. Tim Roughgarden, 'Algorithms Illuminated' (books 1 through 3), Soundlikeyourself Publishing, 2017,18,19 respectively.

R3. AV Aho, J Hopcroft, JD Ullman, 'The Design and Analysis of Algorithms', Addison-Wesley, 1974.

Web Based Resources and E-books:

W1. NPTEL: https://onlinecourses.nptel.ac.in/noc19_cs47/preview

W2. https://presiuniv.knimbus.com/user#/searchresult?searchId=eBook& curPage=0&layout=grid &sorFieldId=none&topresult=false&content=*cloud*

SPECIFIC GUIDELINES TO STUDENTS:

1. **Students are instructed to attend classes regularly.**
2. **Students are instructed to strictly adhere for the assignment deadlines.**
3. **Students are instructed to actively participate in seminar, flipped class, and in any other class room discussions.**
4. **Students are instructed to maintain a running notes for the course.**

COURSE SCHEDULE:

TABLE 4: COURSE BROAD SCHEDULE				
Sl. No.	**ACTIVITY**	**PLANNED STARTING DATE**	**PLANNED CONCLUDING DATE**	**TOTAL NUMBER OF PERIODS**
01	**Over View of the course**	**16-02-2023**	**16-02-2023**	**1**
02	**Module : 01**	**17-02-2023**	**02-03-2023**	**6**
02	**Module: 02**	**06-03-2023**	**30-03-2023**	**11**
03	**Assignment/any other activity /Guest Lecture**	**31-03-2023**	**31-03-2023**	**1**
04	**Module:03**	**03-04-2023**	**10-04-2023**	**4**
05	**Mid Term**	**12-04-2023**	**15-04-2023**	
06	**Module :03 cont..**	**17-04-2023**	**25-04-2023**	**4**
07	**Module:04**	**26-04-2023**	**15-05-2023**	**8**
08	**Assignment**	**16-05-2023**	**16-05-2023**	**1**
09	**Module:05**	**18-05-2023**	**30-05-2023**	**7**
10	**Quiz**	**31-05-2023**	**31-05-2023**	**1**
11	**End Term**			

DETAILED SCHEDULE OF INSTRUCTION:

TABLE 5: DETAILED COURSE SCHEDULE/ LESSON PLAN				
Session no	**TOPIC**	**SUBTOPIC**	**CO Number**	**Reference**
L1	Introduction to Algorithms	Program Integration, Course Integration, Algorithm Design and efficiency	CO1	T1, R1, R2
L2	Introduction to Algorithms	Measuring of running time of algorithms	CO1	T1
L3	Introduction to Algorithms	Asymptotic Growth and Notations	CO1	T1
L4	Introduction to Algorithms	Asymptotic Growth and Notations	CO1	T1
L5	Introduction to Algorithms	Recurrences--Masters method	CO1	T1
L6	Introduction to Algorithms	Recurrences--Masters method, Revision of Module 1	CO1	T1
END OF MODULE 1				
L7	Review of Searching and Sorting techniques	Course Integration,Insertion sort	CO2	T1, R1
L8	Review of Searching and Sorting techniques	Merge Sort	CO2	T1, R1, R2
L9	Review of Searching and Sorting techniques	Quicksort,Heapsort	CO2	T1, R1, R2
L10	Review of Searching and Sorting techniques	Lower bound of comparison-based sorting, non-comparison-based sorting:	CO2	T1, R1, R2

		Radix sort.		
L11	Review of Searching and Sorting techniques	Review of Linear Search and Binary Search	CO2	T1, R1, R2
L12	Review of Searching and Sorting techniques	Hashing and hash tables	CO2	T2,R1, R2
L13	Review of Searching and Sorting techniques	Hashing and hash tables	CO2	T2,R1, R2
L14	Review of Searching and Sorting techniques	Divide and Conquer: Examples	CO2	R1, R2
L15	Review of Searching and Sorting techniques	Divide and Conquer: Examples	CO2	T1, R1, R2
L16	Review of Searching and Sorting techniques	Strassen's Matrix multiplication	CO2	T1, R1
L17	Review of Searching and Sorting techniques	Revision of Module 2	CO2	T1, R1
END OF MODULE 2				
L18	Greedy Algorithms	Course Integration, Introduction	CO3	T1, R1, R2
L19	Greedy Algorithms	Fractional Knapsack Problem	CO3	R1, R2
L20	Greedy Algorithms	Fractional Knapsack Problem	CO3	T2, T1, R2
L21	Greedy Algorithms	Minimal Spanning Tree: Prim's Algorithm	CO3	T2, R2
L22	Greedy Algorithms	Kruskal's Algorithm	CO3	T2, R2
L23	Greedy Algorithms	Problems on Prims and Kruskals Algorithm	CO3	T1,T2
L24	Greedy Algorithms	Single-source Shortest Path: Dijkstra's Algorithm	CO3	T1,T2
L25	Greedy Algorithms	Huffman Codes, Revision	CO3	T1, T2
END OF MODULE 3				
L26	Dynamic Programming	Course Integration Introduction with examples	CO4	T1, T2
L27	Dynamic Programming	Principles of Memorization	CO4	T1, T2
L28	Dynamic Programming	0-1 Knapsack Problem	CO4	T1, T2
L29	Dynamic Programming	Bellman-Ford algorithm	CO4	T1, T2
L30	Dynamic Programming	Floyd-Warshall's Algorithms	CO4	T1, T2
L31	Dynamic Programming	Floyd-Warshall's Algorithms	CO4	T1, T2
L32	Dynamic Programming	Optimal Binary Search Trees	CO4	T1, T2
L33	Dynamic Programming	Chain Matrix Multiplication, Revision	CO4	T1, T2
END OF MODULE 4				
L34	Complexity Classes and Heuristics	Introduction Complexity classes: P, NP, and NP-Complete Problems	CO5	T1, T2,R3
L35	Complexity Classes and Heuristics	Complexity classes: P, NP, and NP-Complete Problems	CO5	T1, T2,R3
L36	Complexity Classes and Heuristics	Complexity classes: P, NP, and NP-Complete Problems	CO5	T1, T2,R3
L37	Complexity Classes and	Backtracking: n-Queens	CO5	T1, T2,R3

	Heuristics			
L38	Complexity Classes and Heuristics	Backtracking: n-Queens	CO5	T1, T2,R3
L39	Complexity Classes and Heuristics	Branch and bound: Travelling Salesman Problem.	CO5	T1, T2,R3
L40	Complexity Classes and Heuristics	Branch and bound: Travelling Salesman Problem, Revision	CO5	T1, T2,R3
END OF MODULE 5				
L41	Module 1,2	Revision of ALL MODULES	CO1,CO2	
L42	Module3,4,5	Revision of ALL MODULES	CO3,CO4,CO5	

ASSESSMENT SCHEDULE:

TABLE 6 ASSESSMENT SCHEDULE							
Sl.no	**Assessment type**	**Contents**	**Course outcome Number**	**Duration In Hours**	**Marks**	**Weightage**	**Venue, DATE &TIME**
1	**Assignment**	**Module 1,2**	**CO1,CO2**	**-**	**20**	**10%**	**March 4th Week 2023**
2	**Mid Term**	**Module 1, 2 and Module 3-50%**	**CO1,CO2,CO3**	**1 hour 30 minutes**	**50**	**25%**	**Wednesday, 12th April 2023 to Saturday, 15th April 2023**
3	**Assignment-Review of digital / e-resources from Pres. Univ. link given in the References Section**	**Module 2,3,4**	**CO1,CO2, CO3,CO4**	**-**	**20**	**10%**	**May First Week 2023**
4	**Quiz**	**Module 4,5**	**CO4,CO5**	**1 hour**	**10**	**5%**	**May Last Week 2023**
5	**End Term**	**Module1,2,3,4,5**	**CO1,CO2,CO3, CO4,CO5**	**3 hour**	**100**	**50%**	**AS per COE timetable**

COURSE CLEARANCE CRITERIA:

-"AS PER ACADEMIC REGULATIONS OF THE UNIVERSITY")

MAKEUP EXAM POLICY:

"AS PER ACADEMIC REGULATIONS OF THE UNIVERSITY")

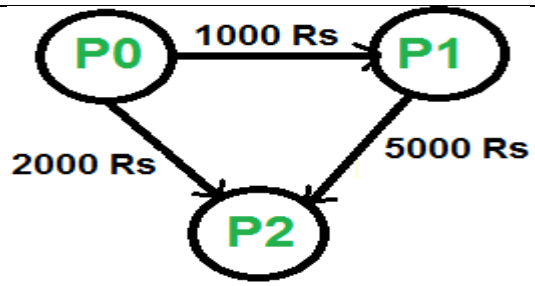
CONTACT TIMINGS IN THE CHAMBER FOR ANY DISCUSSIONS:

Wednesday 2.20pm to 4pm

[Each faculty while sharing the handout has to mention the free hour of them checking the section timetable]

SAMPLE THOUGHT PROVOKING QUESTIONS:

TABLE 7: SAMPLE THOUGHT PROVOKING QUESTIONS				
SL NO	**QUESTION**	**MARKS**	**COURSE OUTCOME NO.**	**BLOOM'S LEVEL**
1	In a candy store there are N different types of candies available, the prices of all the N different types of candies are provided to you. What is the minimum amount of money you have to spend to buy all the N different candies. what is the maximum amount of money you have to spend to buy all the N different candies for the amount given to you.	2	CO4	L2
2	Given a sorted array A, size N, of integers; every element appears twice except for one. Find that element in linear time complexity and without using extra memory.	5	CO2	L2
3	The problem is to find shortest distances between every pair of vertices in a given edge weighted directed Graph. The Graph is represented as Adjacency Matrix, and the Matrix denotes the weight of the edges (if it exists) else INF.	5	CO3	L3
4	Given a number of friends who have to give or take some amount of money from one another. Design an algorithm by which the total cash flow among all the friends is minimized.	5	CO4	L3

	 P0 has to pay 1000 Rs to P1 P0 also has to pay 2000 Rs to P2 P1 has to pay 5000 Rs to P2.			
5	A Student wants to search for a book in the library. Suggest the best searching technique and justify your answer in terms of time complexity.	5	CO2	L2

TARGET SET FOR COURSE OUTCOME ATTAINMENT:

TABLE 8: TARGET SET FOR ATTAINMENT OF EACH CO and ATTAINMENT ANALYSIS AFTER RESULTS						
Sl.no	**C.O. No.**	**Course Outcomes**	**Threshold Set for the CO**	**Target set for attainment in percentage**	Actual C.O. Attainment In Percentage	Remarks on attainment &Measures to enhance the attainment
01	CO1	Identify the efficiency of a given algorithm	75	70		
02	CO2	Employ divide and conquer approach to solve a problem	65	60		
03	CO3	Illustrate dynamic programming approach to solve a given problem	65	60		
04	CO4	Solve problem using the greedy method	65	60		
05	CO5	Discuss the techniques to	65	60		

		solve a real-world problem based on its complexity classes				

Signature of the course Instructor In-Charge (s)

APPROVAL:

This course has been duly verified Approved by the D.A.C.

Signature of the Chairperson D.A.C.

Name and signature of the Instructor In-Charge (s) :

Name and signature of the DAC Chairperson:

BLOOM'S TAXONOMY SAMPLE VERBS

Learning Outcomes Verbs at Each Bloom Taxonomy Level to be used for writing the course Outcomes.

TABLE 9: REFERENCE SAMPLES OF BLOOMS TAXONOMY VERBS		
Cognitive Level	Illustrative Verbs	Definitions
Knowledge	arrange, define, describe, duplicate, identify, label, list, match, memorize, name, order, outline, recognize, relate, recall, repeat, reproduce, select, state	remembering previously learned information
Comprehension	classify, convert, defend, discuss, distinguish, estimate, explain, express, extend, generalize, give example(s), identify, indicate, infer, locate, paraphrase, predict, recognize, rewrite, report, restate, review, select, summarize, translate	grasping the meaning of information
Application	apply, change, choose, compute, demonstrate, discover, dramatize, employ, illustrate, interpret, manipulate, modify, operate, practice, predict, prepare, produce, relate schedule, show, sketch, solve, use write	applying knowledge to actual situations
Analysis	analyze, appraise, breakdown, calculate, categorize, classify, compare, contrast, criticize, derive, diagram, differentiate, discriminate,	breaking down objects or ideas into simpler parts and seeing how the parts relate and are organized

	distinguish, examine, experiment, identify, illustrate, infer, interpret, model, outline, point out, question, relate, select, separate, subdivide, test	
Synthesis	arrange, assemble, categorize, collect, combine, comply, compose, construct, create, design, develop, devise, explain, formulate, generate, plan, prepare, propose, rearrange, reconstruct, relate, reorganize, revise, rewrite, set up, summarize, synthesize, tell, write	rearranging component ideas into a new whole
Evaluation	appraise, argue, assess, attach, choose, compare, conclude, contrast, defend, describe, discriminate, estimate, evaluate, explain, judge, justify, interpret, relate, predict, rate, select, summarize, support, value	making judgments based on internal evidence or external criteria