CURRICULUM AND DETAILED SYLLABI

FOR

B.E DEGREE (Computer Science and Engineering) PROGRAMME

FOURTH SEMESTER

FOR THE STUDENTS ADMITTED FROM THE

ACADEMIC YEAR 2018 - 2019 ONWARDS

THIAGARAJAR COLLEGE OF ENGINEERING
(A Government Aided ISO 9001-2008 certified
Autonomous Institution affiliated to Anna University)

MADURAI – 625 015, TAMILNADU

Phone: 0452 – 2482240, 41
Fax: 0452 2483427
Web: www.tce.edu

Passed in Board of Studies Meeting on 11.05.2019

Approved in 58th Academic Council Meeting on 15.06.2019

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015
B.E Degree (Computer Science and Engineering) Programme
COURSES OF STUDY
(For the candidates admitted from 2018 -19 onwards)

FOURTH SEMESTER

Course Code	Name of the Course	Category	No. of Hours / Week			Credits
			L	T	P	
THEORY						
18CS410	Discrete Mathematics	BS	2	1	-	3
18CS420	Design and Analysis of Algorithms	PC	3	-	-	3
18CS430	System Software and Operating Systems	PC	3	-	-	3
18CS440	Database Management Systems	PC	2	1	-	3
18XXFX0	Foundation Elective - I	FE	3	-	-	3
19CS490	Project Management	HSS	3	-	-	3
THEORY CUM PRACTICAL						
18EG460	Professional Communication	HSS	-	1	2	2
PRACTICAL						
18CS470	System software and Operating Systems Lab	PC	-	-	2	1
18CS480	Algorithms Lab	PC	-	-	2	1
MANDATORY AUDIT COURSE						
18CHAB0	Constitution of India	AC	2	-	-	-
Total			**18**	**3**	**6**	**22**

AC : Audit Course
BS : Basic Science
HSS : Humanities and Social Science
ES : Engineering Science
PC : Program Core
L : Lecture
T : Tutorial
P : Practical

Note:

1 Hour Lecture is equivalent to 1 credit

1 Hour Tutorial is equivalent to 1 credit

2 Hours Practical is equivalent to 1 credit

THIAGARAJAR COLLEGE OF ENGINEERING: MADURAI – 625 015
B.E Degree (Computer Science and Engineering) Programme
SCHEME OF EXAMINATIONS
(For the candidates admitted from 2018-19 onwards)

FOURTH SEMESTER

S.No.	Course Code	Name of the Course	Duration of Terminal Exam. in Hrs.	Marks			Minimum Marks for Pass	
				Continuous Assessment	Terminal Exam *	Max. Marks	Terminal Exam	Total
THEORY								
1	18CS410	Discrete Mathematics	3	50	50	100	25	50
2	18CS420	Design and Analysis of Algorithms	3	50	50	100	25	50
3	18CS430	System Software and Operating Systems	3	50	50	100	25	50
4	18CS440	Database Management Systems	3	50	50	100	25	50
5	18XXFX0	Foundation Elective - I	3	50	50	100	25	50
6	19CS490	Project Management	3	50	50	100	25	50
THEORY CUM PRACTICAL								
7	18EG460	Professional Communication	3	50	50	100	25	50
PRACTICAL								
8	18CS470	System Software and Operating Systems Lab	3	50	50	100	25	50
9	18CS480	Algorithms Lab	3	50	50	100	25	50
MANDATORY AUDIT COURSE								
10	18CHAB0	Constitution of India	-	50	50	100	25	50

* Terminal Examination will be conducted for maximum marks of 100 and subsequently be reduced to 50 marks for the award of terminal examination marks

18CS410	DISCRETE MATHEMATICS

Category	L	T	P	Credit
BS	2	1	0	3

Preamble

A course in discrete mathematics teaches students how to work with discrete structures, which are the abstract mathematical structures used to represent discrete objects and relationships between these objects. These discrete structures include logic, predicate calculus and sets. An important problem–solving skill is the ability to count or enumerate objects. The discussion of enumeration in this course begins with basic techniques of counting. The general counting methods involve permutations and combinations. These methods are very useful in constructing computer programs and in mastering many theoretical topics of computer science. Recurrence relations are one of the simplest ways to solve counting problems. The methods for solving recurrence relations appeared originally in the development of the theory of difference equations, cousins of differential equations.

Prerequisite

Nil

Course Outcomes

On the successful completion of the course students will be able to

CO Number	Course Outcome Statement	Weightage*** in %
CO1	Outline an equivalent logical proposition for a real world statement by applying predicates and quantifiers and Interpret	15
CO2	Convert the given normal form into its Principle conjunctive normal forms, Principle disjunctive normal forms by applying required quantifiers	10
CO3	Apply logic rules of inference to check the validity of the predicate calculus statements and to prove theorems	25
CO4	Apply the concepts of sets, functions and relations to solve the given problem.	20
CO5	Apply basic counting techniques to solve combinatorial Problems.	10
CO6	Construct the recurrence relation for a given engineering problem and solve the recurrence relation.	20

CO Mapping with CDIO Curriculum Framework

CO #	TCE Proficiency Scale	Learning Domain Level			CDIO Curricular Components (X.Y.Z)
		Cognitive	Affective	Psychomotor	
CO1	TPS3	Apply	Value		1.1.1, 2.4.4, 3.2.2
CO2	TPS3	Apply	Value		1.1.1, 2.4.4
CO3	TPS3	Apply	Value		1.1.1, 2.4.4
CO4	TPS3	Apply	Value		1.1.1, 2.4.4
CO5	TPS3	Apply	Value		1.1.1, 2.4.4
CO6	TPS3	Apply	Value		1.1.1, 2.4.4, 2.4.7

Mapping with Programme Outcomes and Programme Specific Outcomes

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	S	M	L							L			M		
CO2	S	M	L							L			M		

CO3	S	M	L							L			M		
CO4	S	M	L							L			M		
CO5	S	M	L							L			M		
CO6	S	M	L							L			M		

S- Strong; M-Medium; L-Low

Assessment Pattern: Cognitive Domain

Cognitive Levels	Continious Assessment Tests			Assignment			Terminal Examinations
	1	2	3	1	2	3	
Remember	10	10	10	-	-	-	-
Understand	30	30	30	-	-	-	30
Apply	60	60	60	100	100	100	70
Analyse							
Evaluate							
Create							

Assessment Pattern: Psychomotor

Psychomotor Skill	Miniproject /Assignment/Practical Component
Perception	
Set	
Guided Response	
Mechanism	
Complex Overt Responses	
Adaptation	
Origination	

Sample Questions for Course Outcome Assessment**

Course Outcome 1(CO1):

1. Express the statement “Every student in this class has studied calculus” using predicates and quantifiers.
2. Show that $\forall x(P(x) \wedge Q(x))$ and $\forall x P(x) \wedge \forall x Q(x)$ are logically equivalent
3. Translate into English the statement $\forall x \forall y((x > 0) \wedge (y < 0) \rightarrow (xy < 0))$,where the domain for both variables consists of all real numbers.

Course Outcome 2(CO2):

1. Find the principal disjunctive normal form (PDNF) of a Boolean expression $((p \wedge q) \rightarrow r) \vee ((p \wedge q) \rightarrow \neg r)$.
2. Obtain the PDNF of $(\neg P \vee \neg Q) \rightarrow (P \Leftrightarrow \neg Q)$
3. Obtain PCNF for A : $(\neg P \rightarrow R) \wedge ((Q \rightarrow P) \wedge (P \rightarrow Q))$.

Course Outcome 3(CO3):

1. Show that the premises “It is not sunny this afternoon and it is colder than yesterday,” “We will go swimming only if it is sunny,” “If we do not go swimming, then we will take a canoe trip,” and “If we take a canoe trip, then we will be home by sunset” lead to the conclusion “We will be home by sunset.”

2. Prove that the sum of two rational numbers is rational. (Note that if we include the implicit quantifiers here, the theorem we want to prove is "For every real number *r* and every real number *s*, if *r* and *s* are rational numbers, then *r* + *s* is rational.)
3. Formulate a conjecture about the final decimal digit of the square of an integer and prove your result.

Course Outcome 4(CO4):

1. Suppose that *R* is the relation on the set of strings of English letters such that *aRb* if and only if *l(a)* = *l(b)*, where *l(x)* is the length of the string *x*. Is *R* an equivalence relation?
2. Let *R* be the relation on the set of people such that *xRy* if *x* and *y* are people and *x* is older than *y*. Show that *R* is not a partial ordering
3. Draw the Hasse diagram for the partial ordering {*(A,B)* | *A* ⊆ *B*} on the power set *P(S)* where *S* = {*a, b, c*}.

Course Outcome 5 (CO5):

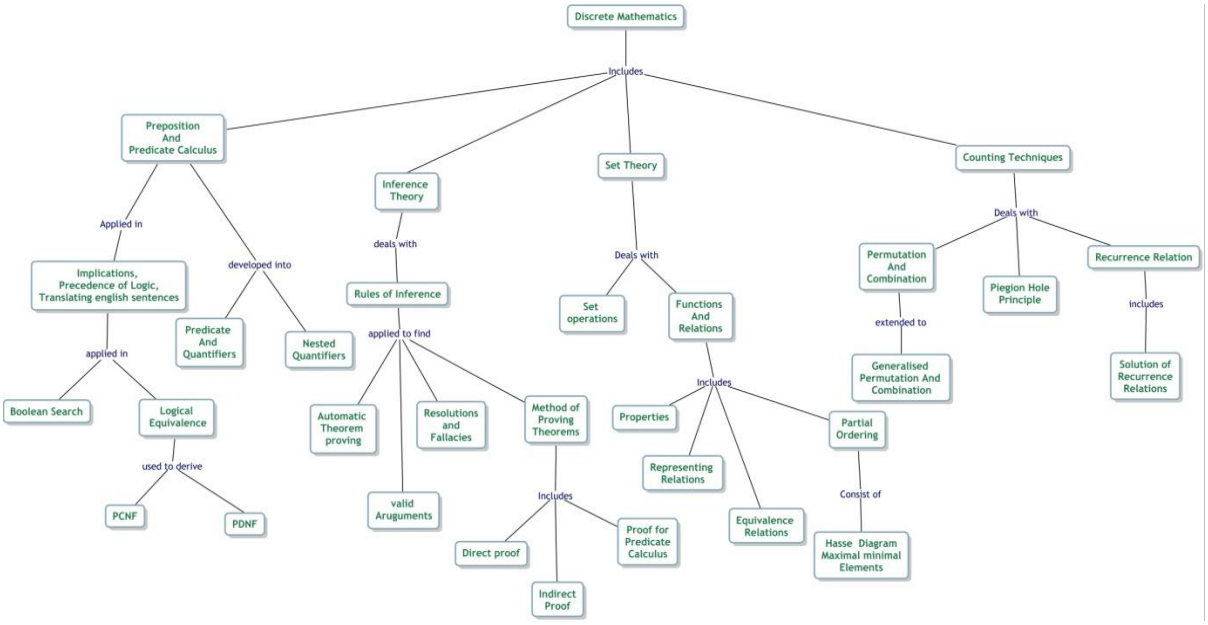
1. Suppose that "I Love New Jersey" T-shirts come in five different sizes: S, M, L, XL, and XXL. Further suppose that each size comes in four colors, white, red, green, and black, except for XL, which comes only in red, green, and black, and XXL, which comes only in green and black. How many different shirts does a souvenir shop have to stock to have at least one of each available size and color of the T-shirt?
2. What is the least number of area codes needed to guarantee that the 25 million phones in a state can be assigned distinct 10-digit telephone numbers? (Assume that telephone numbers are of the form *NXX-NXX-XXXX*, where the first three digits form the area code, *N* represents a digit from 2 to 9 inclusive, and *X* represents any digit.)
3. How many ways are there to select four pieces of fruit from a bowl containing apples, oranges, and pears if the order in which the pieces are selected does not matter, only the type of fruit and not the individual piece matters, and there are at least four pieces of each type of fruit in the bowl?

Course Outcome 6 (CO6):

1. A computer system considers a string of decimal digits a valid codeword if it contains an even number of 0 digits. For instance, 1230407869 is valid, whereas 120987045608 is not valid. Let a^n be the number of valid *n*-digit code words. Find a recurrence relation for a^n.
2. Find an explicit formula for the Fibonacci numbers.
3. Give a big-*O* estimate for the number of comparisons used by a binary search.

.

Concept Map



Syllabus

Statement and Predicate Calculus: Propositions, Implications, Precedence of logical operators, Translating English sentences, System specifications, Boolean search, Logic and bit operators, Propositional Equivalence, Logical equivalence, Principle conjunctive normal forms, Principle disjunctive normal forms, Predicates and Quantifiers, Nested Quantifiers.

Inference Theory: Rules of inference, Valid arguments, Resolution and Fallacies, Rules of Inference for quantifiers, Methods of proving theorems, Automatic Theorem proving.

Set Theory: Set operations, Functions, Relations and Properties, Representing relations, Equivalence relations, Partial orderings- Introduction - Hasse diagrams- Maximal and Minimal Elements.

Counting Techniques: The basics of counting, The Pigeonhole Principle, Permutation and Combinations, Binomial Coefficients, Generalized Permutation and Combinations, Generating Permutation and Combinations, Recurrence Relations, Solving Recurrence Relations, Generating Functions.

Learning Resources

1. Kenneth H. Rosen, “Discrete Mathematics and its Applications”, 7th edition, Tata McGraw Hill, 2012
   Statement and Predicate Calculus : Section 1.1, 1.2, 1.3, 1.4, 1.5
   Inference Theory : Section 1.6, 1.7, 1.8
   Set : Section 2.2, 2.3, 9.1, 9.3, 9.5, 9.6
   Counting Techniques : Section 6.1, 6.2, 6.3, 6.4, 6.5, 6.6, 8.1, 8.2, 8.3, 8.4
2. T.P.Tremblay and R.Manohar , "Discrete Mathematical Structures with application to Computer Science”,Tata McGraw Hill, 2002.
   Statement and Predicate Calculus : Section 1.3 (for PCNF and PDNF only)
   Inference Theory : Section 1.4.4 (Automatic theorem proof only)
3. Alan Tucker, “Applied Combinatorics", John Wiley & Sons, Incorporated, 2012.
4. Liu, C.L. “Introduction to Combinatorial Mathematics”, McGraw Hill Book Company, New York, 1968.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Lectures	Course Outcomes
1.	**Logic**		
1.1	Propositions, Implications, Precedence of logical operators, Translating English sentences	1	CO1
	Tutorial	1	CO1
1.2	Boolean search, Logic and bit operators	1	CO1
1.3	Logical equivalence	1	CO1
1.4	Principle Conjunctive normal forms, Principle Disjunctive normal forms.	1	CO2
	Tutorial	1	CO2
1.5	Predicates and Quantifiers	1	CO1
1.6	Nested Quantifiers	1	CO1
	Tutorial	1	CO1
2	**Inference theory**		
2.1	Rules of Inference	1	CO3
2.2	Valid Arguments	1	CO3
	Tutorial	1	CO3
2.3	Resolution and Fallacies	1	CO3
2.4	Rules of Inference for Quantifiers	1	CO3
	Tutorial	1	CO3
2.5	Methods of Proving theorems	1	CO3
	Tutorial	1	CO3
2.6	Automatic proving theorem	1	CO3
3	**Set Theory**		
3.1	Set Operations	1	CO4
3.2	Functions	1	CO4
	Tutorial	1	CO4
3.3	Relations and Properties	1	CO4
3.4	Representing relations	1	CO4
	Tutorial	1	CO4

Module No.	Topic	No. of Lectures	Course Outcomes
3.5	Equivalence relations	1	CO4
3.6	Partial ordering- Introduction, Hasse diagrams, Maximal minimal elements	1	CO4
	Tutorial	1	CO4
4	**Counting Techniques**		
4.1	The basics of counting, The Pigeonhole Principle	1	CO5
4.2	Permutation and Combination, Binomial Coefficients	1	CO5
	Tutorial	1	CO5
4.3	Generalized permutation and combinations, Generating permutation and combinations	1	CO5
	Tutorial	1	CO5
4.4	Recurrence Relations,	1	CO6
4.5	Solving Recurrence relations,	1	CO6
4.6	Generating Functions	1	CO6
	Tutorial	1	CO6
	Total	**36**	

Course Designers:

1. Dr.A.Anitha anithavalli@tce.edu
2. Ms. H.Sri Vinodhini srivinodhini@tce.edu

18CS420	

	DESIGN AND ANALYSIS OF ALGORITHMS

Category	L	T	P	Credit
PC	3	0	0	3

Preamble

Algorithm design and analysis provide the theoretical backbone of computer science. On completion of this course students will be able to:

i) Determine the asymptotic time complexity of algorithms
ii) Write rigorous correctness proofs for algorithms
iii) Use different paradigms of problem solving to illustrate efficient ways of solving a given problem

Prerequisite

- Problem Solving using Computers
- Data Structures and Algorithms
- Data Structures Lab

Course Outcomes

On the successful completion of the course, students will be able to

CO Number	Course Outcome Statement	Weightage*** in %
CO1	Determine the running times of algorithms using asymptotic analysis and explain the significance of NP-completeness	15
CO2	Describe the divide-and-conquer paradigm and solve recurrences describing the performance of divide-and-conquer algorithms.	15
CO3	Construct graph-based algorithms to solve engineering problems.	20
CO4	Apply design principles for developing solutions using greedy algorithm approaches.	20
CO5	Analyse the algorithms and design techniques of dynamic programming to solve real world problems and mathematically evaluate the quality of the solutions.	15
CO6	Construct algorithms using branch and bound to solve any given problem.	15

CO Mapping with CDIO Curriculum Framework

CO #	TCE Proficiency Scale	Learning Domain Level			CDIO Curricular Components (X.Y.Z)
		Cognitive	Affective	Psychomotor	
CO1	TPS3	Understand	Respond	Guided Response	1.2

CO2	TPS3	Apply	Value	Mechanism	1.2,2.1.1
CO3	TPS3	Apply	Value	Mechanism	1.2,2.1.1
CO4	TPS3	Apply	Value	Mechanism	1.2,2.1.1
CO5	TPS3	Analyze	Organise	Complex overt Responses	1.2,2.1.1
CO6	TPS2	Apply	Value	Mechanism	1.2,2.1.1

Mapping with Programme Outcomes and Programme Specific Outcomes

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	M	L											L		
CO2	S	M	L						L	M		M	M		M
CO3	S	M	L						L	M		M	M		M
CO4	S	M	L						L	M		M	M		M
CO5	S	S	M	L					L	M		M	S		M
CO6	S	M	L						L	M		M	M		M

S- Strong; M-Medium; L-Low

Assessment Pattern: Cognitive Domain

Cognitive Levels	Continuous Assessment Tests			Assignment			Terminal Examination
	1	2	3	1	2	3	
Remember	20	20	10	-	-	-	20
Understand	40	40	20	50	50	-	30
Apply	40	40	50	50	50		50
Analyse	-	-	20	-	-	-	-
Evaluate	-	-	-	-	-	-	-
Create	-	-	-	-	-	-	-

Assessment Pattern: Psychomotor

Psychomotor Skill	Assignment-3
Perception	-
Set	-
Guided Response	50
Mechanism	50

Complex Overt Responses	-
Adaptation	-
Origination	

Course Outcome 1 (CO1):

1. Consider the searching problem: given an array A[1…n] and a value v output an Index i such that v = A[i] or the special value ϕ if v does not appear in A. If the array J is sorted, we can perform a binary search: compare P with the midpoint of the array and repeat the search on one half of array, eliminating the other half from further consideration.
   (a) Construct a pseudocode for binary search as a recursive procedure.
   (b) Rewrite your binary search procedure in an iterative style.
   (c) Formally state pre- and post-conditions for your iterative procedure. Deduce a loop invariant, and illustrate that your procedure is correct.

2. Demonstrate that Bubble Sort works by proving that
   a. It terminates.
   b. It sorts the set of numbers provided as an input.

   Deduce a loop-invariant to prove the correctness of the selection sort algorithm
   Let X1, . . . ,Xn be {0, 1}-valued random variables such that Xi = 0 with probability 1 − pi and Xi = 1 with probability pi . Given probabilities p1, . . . ,pn, show how to compute probabilities of events P i Xi = m for all m ∈ [0, n], using a divide-and-conquer approach. Analyze the running time of your algorithm.
3. Let X be an NP-Complete problem. Consider a decision problem Z Є NP such that X $\leq_T^P$ Z. Then defend that Z is also NP-Complete.

Course Outcome 2 (CO2):

1. Although merge sort runs in Θ(n log2 n) worst-case time and insertion sort runs in Θ(n2) worst-case time, the constant factors in insertion sort make it faster for small n. Thus, it makes sense to use insertion sort within merge sort when sub-problems become sufficiently small. Consider a modification to merge sort in which n/k sub-lists of length k are sorted using insertion sort and then merged using the standard merging mechanism, where k is a value to be determined.
   a) Show that the n/k sub-lists, each of length k, can be sorted by insertion sort in Θ(nk) worst-case time.
   b) Show that the sub-lists can be merged in Θ(n log2 (n/k) worst-case time.
2. The following code-fragment implements Horner's rule for evaluating a polynomial $P(x) = \sum_{k=0}^{n} a_k x^k$

```
y = 0
for i = n down to 0
    y = a_i + x * y
```

   a) In terms of Θ notation, calculate the running time of this code fragment for Horner's rule
   b) Construct a pseudo-code to implement the naive polynomial-evaluation algorithm that computes each term of the polynomial from scratch. Calculate the running time of this algorithm. Compare it to the Horner's rule.

Course Outcome 3(CO3)

1. Show that a depth first search of an undirected graph 'G' can be used to identify the connected components of 'G' and that the depth first forest contains as many trees as 'G' has connected components. More precisely, show how to modify depth-first-search so that each vertex v is assigned an integer label cc[v] between 1 and k, where 'k' is the

number of connected components of 'G' such that cc[u] = cc[v] if and only if u and v are in the same connected component.

2. We are given a directed graph G = (V, E) on which each edge (u,v) $\in$ E has an associated value r(u,v), which is a real number in the range $0 \leq r(u,v) \leq 1$ that represents the reliability of a communication channel from vertex u to vertex v. We interpret r(u,v) as the probability that the channel from u to v will not fail, and we assume that these probabilities are independent. Construct an efficient algorithm to find the most reliable path between two given vertices.

3. Determine an algorithm to detect cycles in a directed graph G(V, E) in O(|V| + |E|) time.

Course Outcome 4 (CO4):

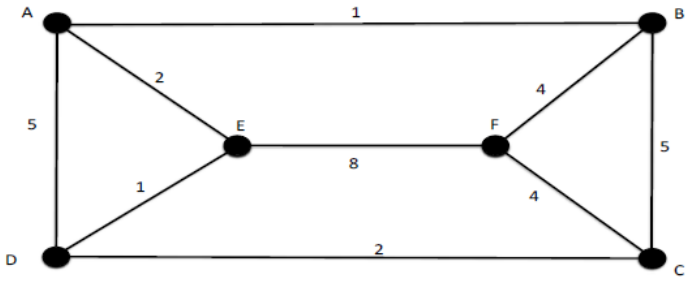
1. Defend that the greedy algorithm that solves the scheduling problem with the goal of minimizing the time spent by the customers in a system is optimal, if it adopts the following greedy strategy:
   "At each step add to the end of the schedule the customer requiring the least service time among those who remain"
2. Suppose that you have a median(S) algorithm, which finds the median element in the sequence S in O(n) time. Using this median(S) algorithm, construct a simple O(n) – linear time algorithm that solves the selection problem Select(S, k).
3. Suppose instead of running Dijkstra's algorithm till the priority queue 'Q' becomes empty, we run it as long as |Q| > 1. This change will cause the 'while' loop in Dijkstra's algorithm to execute |V| - 1 times instead of |V| times. Analyze whether the proposed algorithm is correct

Course Outcome 5 (CO5):

1. Write the pseudo code to find the optimal parenthesization of a matrix-chain product. Using the algorithm, find the optimal parenthesization of a matrix-chain product whose sequence of dimensions is (5,10,3,12,5)
2. Suppose we wish to find a shortest path from vertex i to vertex j. Let Ai be the vertices adjacent from vertex i. Which of the vertices in Ai should be the second vertex on the path? There is no way to make a decision at this time and guarantee that future decisions leading to an optima lsequence can be made. If on the other hand we wish to find a shortest path from vertex i to all other vertices in G ,then at each step, a correct decision can be made.
3. Deduce that backtracking algorithm solves the 2-SAT problem in polynomial time.

Course Outcome 6(CO6):

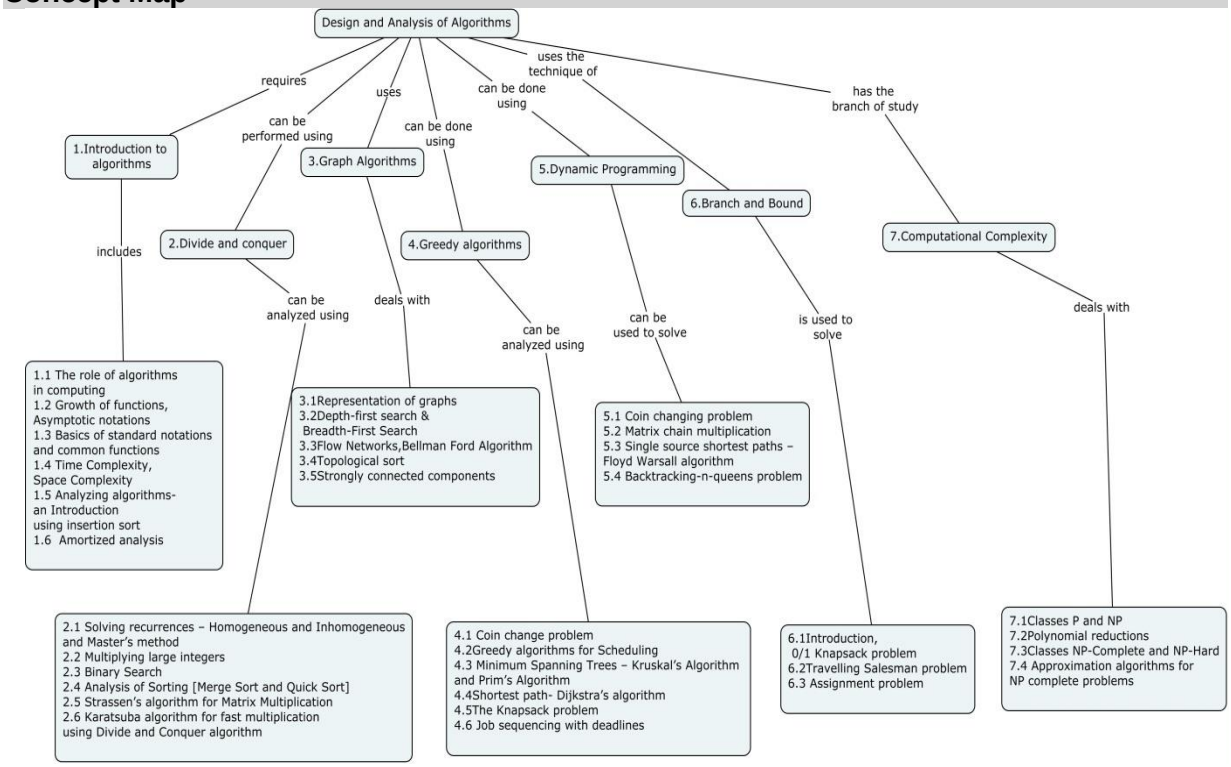
1. Apply the Branch and Bound Method (using the MST-based bounding scheme) seen in class for the TSP to solve the following instance.



2. Consider three jobs to be assigned to three machines. The cost for each combination is shown in the table below. Determine the minimal job – machine combinations

Job	Machine			
	1	2	3	a_i
1	5	7	9	1
2	14	10	12	1
3	15	13	16	1
b_j	1	1	1	

Concept Map



Syllabus

Introduction to algorithms: The role of algorithms in computing, Growth of functions, Asymptotic notations, Basics of standard notations and common functions ,Time Complexity, Space Complexity , Analyzing algorithms- an Introduction using insertion sort- Amortized analysis. **Divide and Conquer**: Solving recurrences – Homogeneous and Inhomogeneous, Master's method, Binary Search, Analysis of Sorting [Merge Sort, Quick Sort], Strassen's algorithm for Matrix Multiplication, Karatsuba algorithm for fast multiplication using Divide and Conquer algorithm. **Graph Algorithms**: Representation of graphs, Depth-first search & Breadth-First Search, Flow Networks, Bellman Ford Algorithm, Topological sort, strongly connected components. **Greedy Algorithms**: Coin change problem, Minimum Spanning Trees – Kruskal's Algorithm and Prim's Algorithm, Shortest path- Dijkstra's algorithm, The Knapsack problem ,Job sequencing with deadlines. **Dynamic Programming**: Coin changing problem, Matrix chain multiplication, Single source shortest paths – Floyd Warshall algorithm, Backtracking-n-queens problem. **Branch and Bound –** Introduction **-** 0/1 Knapsack problem**,** Travelling Salesman problem, Assignment problem. **Computational Complexity**: Classes P and NP, Polynomial reductions, Classes NP-Complete and NP-Hard, Approximation algorithms for NP complete problems.

Learning Resources

1. Charles E. Leiserson, Thomas H. Cormen, Ronald L. Rivest, Clifford Stein, Introduction to Algorithms, Third edition, PHI, 2010.
2. Gilles Brassard and Paul Bratley - Fundamentals of Algorithmics, PHI, 2009
3. Ellis Horowitz, SartajSahni and SanguthevarRajasekaran, Fundamentals of Computer Algorithms, Second Edition, Universities Press, 2011
4. Steven S. Skiena,The Algorithm Design Manual,Second Edition,Springer,2010.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Lectures	Course Outcome
1.	**Introduction to Algorithms**		
1.1	The role of algorithms in computing	1	CO1
1.2	Growth of functions, Asymptotic notations		CO1
1.3	Basics of standard notations and common functions		CO1
1.4	Time Complexity, Space Complexity	1	CO1
1.5	Analyzing algorithms- an Introduction using insertion sort	1	CO1
1.6	Amortized analysis		CO1
2.	**Divide and Conquer**		
2.1	Solving recurrences – Homogeneous and Inhomogeneous, Master's method	1	CO2
2.2	Multiplying large integers	1	CO2
2.3	Binary Search	1	CO2

Module No.	Topic	No. of Lectures	Course Outcome
2.4	Analysis of Sorting [Merge Sort and Quick Sort]		CO2
2.5	Strassen's algorithm for Matrix Multiplication	1	CO2
2.6	Karatsuba algorithm for fast multiplication using Divide and Conquer algorithm	1	
3	**Graph Algorithms**		
3.1	Representation of graphs	1	CO3
3.2	Depth-first search & Breadth-First Search	2	CO3
3.3	Flow Networks, Bellman Ford Algorithm	2	CO3
3.4	Topological sort	2	CO3
3.5	Strongly connected components	1	CO3
4	**Greedy Algorithms**		
4.1	Coin change problem	2	CO4
4.2	Greedy algorithms for Scheduling	1	CO4
4.3	Minimum Spanning Trees – Kruskal's Algorithm and Prim's Algorithm	2	CO4
4.4	Shortest path- Dijkstra's algorithm	2	CO4
4.5	The Knapsack problem	1	CO4
4.6	Job sequencing with deadlines		
5	**Dynamic programming**		
5.1	Coin changing problem	1	CO5
5.2	Matrix chain multiplication	2	CO5
5.3	Single source shortest paths – Floyd Warsall algorithm	1	CO5
5.4	Backtracking-n-queens problem	1	CO5
6	**Branch and Bound**		
6.1	Introduction, 0/1 Knapsack problem	1	CO6
6.2	Travelling Salesman problem	2	CO6
6.3	Assignment problem	2	CO6
7	**Computational Complexity**		
7.1	Classes P and NP	1	CO1
7.2	Polynomial reductions		CO1
7.3	Classes NP-Complete and NP-Hard	1	CO1
7.4	Approximation algorithms for NP complete problems.		
	Total Hours	**36**	

Course Designers:

1. Raja Lavanya rlit@tce.edu

18CS430	SYSTEM SOFTWARE AND OPERATING SYSTEMS

Category	L	T	P	Credit
PC	3	0	0	3

Preamble

To provide a basic knowledge in the concepts of system software and to understand the working principle of assemblers, functions of loaders, linkers and macro processors. Operating systems is an essential part of any computer-science education. Although this field is undergoing rapid change, as computers are now prevalent in virtually every application, the fundamental concepts remain fairly clear. It provides a clear description of the *concepts* that underlie operating systems. The fundamental concepts and algorithms are based on those used in existing commercial operating systems.

Prerequisite

Computer Organization and Microprocessor

Course Outcomes

On the successful completion of the course students will be able to

CO Number	Course Outcome Statement	Weightage*** in %
CO1	Describe the functions and types of assemblers, linkers and loaders.	8
CO2	Develop single-pass and multi-pass macro processors.	8
CO3	Develop programs using system-calls related to process, memory and file management	17
CO4	Construct solutions for problems related to process scheduling, deadlocks and synchronization in a multi-programmed operating system.	22
CO5	Develop appropriate solutions for memory management considering challenges due to multi-programming and virtual memory.	25
CO6	Construct solutions for problems related to secondary storage management with an understanding of file systems and disk scheduling.	20

CO Mapping with CDIO Curriculum Framework

CO #	TCE Proficiency Scale	Learning Domain Level			CDIO Curricular Components (X.Y.Z)
		Cognitive	Affective	Psychomotor	
CO1	TPS2	Understand	Respond		1.2,2.3.2
CO2	TPS2	Understand	Respond	Guided Response	1.2,2.3.2

CO3	TPS3	Apply	Value	Mechanism	1.2,2.3.2
CO4	TPS3	Apply	Value	Mechanism	1.2,4.5.1
CO5	TPS3	Apply	Value	Mechanism	1.2, 4.5.1
CO6	TPS3	Apply	Value	Mechanism	1.2, 4.5.1

Mapping with Programme Outcomes and Programme Specific Outcomes

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	M	L											L		
CO2	M	L											L		
CO3	S	M	L					L	L	L		L	M		L
CO4	S	M	L					L	L	L		L	M		L
CO5	S	M	L					L	L	L		L	M		L
CO6	S	M	L					L	L	L		L	M		L

S- Strong; M-Medium; L-Low

Assessment Pattern: Cognitive Domain

Cognitive Levels	Continuous Assessment Tests			Assignment			Terminal Examination
	1	**2**	**3**	1	2	3	
Remember	30	10	10	-	-	-	10
Understand	40	30	30	10	10		40
Apply	30	60	60	90	90	100	50
Analyse	-	-	-	-	-	-	-
Evaluate	-	-	-	-	-	-	-
Create	-	-	-	-	-	-	-

Assessment Pattern: Psychomotor

Psychomotor Skill	Miniproject /Assignment/Practical Component
Perception	-
Set	-
Guided Response	15
Mechanism	85
Complex Overt Responses	-
Adaptation	-
Origination	-

Sample Questions for Course Outcome Assessment**

Course Outcome 1(CO1):

1. What are called assembler directives?
2. Explain the data structures used by the assembler.
3. Differentiate the concepts of single pass and multi pass translation.
4. Distinguish absolute loaders and relocating loaders.
5. Illustrate the concept of dynamic linking.

Course Outcome 2(CO2):

1. Illustrate how nested macro calls are processed by macro processor.

2. Write the syntax for macro definition and macro call.
3. Explain the steps involved in the design of macro processor.
4. Compare and Contrast the properties of macro and subroutines.
5. Describe the data structures used by macro processor.

Course Outcome 3(CO3):

1. Using system calls, a program in either C or C++ that reads data from one file and copies it to another file.
2. List the three major activities of an operating system in regard to memory management?
3. State the purpose of system calls?
4. List the various Memory Management schemes?
5. Distinguish between logical and physical address space?
6. Explain internal and external fragmentation of main memory?

Course Outcome 4 (CO4):

1. Assume you have the following jobs to execute with one processor, with the jobs arriving in the order listed here:

i	T(pi)
0	80
1	20
2	10
3	20
4	50

   a. Suppose a system uses FCFS scheduling. Create a Gantt chart illustrating the execution of these processes?
   b. State the turnaround time for process p3?
   c. List the average wait 0time for the processes?

2. Suppose the following jobs arrive for processing at the times indicated, each job will run the listed amount of time.

Jobs	Arrival time	Burst time (in secs.)
1	0.0	8
2	0.4	4
3	1.0	1

   Give Gantt charts illustrating the execution of these jobs using the non preemptive FCFS and SJF scheduling algorithms. Compute the average turn around time and average waiting time of each job for the above algorithms and find the best alternative.
3. Give a solution to the readers-writers problem after explaining its nature?
4. Apply Dijkstra-Haberman algorithm for deadlock avoidance?
5. Implement Simulated Semaphore(Synchronized Produced Consumer Problem)
6. Discuss how file sharing semantics of unix can be implemented. Can processing of the link and unlink commands of unix lead to deadlocks .Discuss how such deadlocks can be avoided

Course Outcome 5 (CO5):

1. Given memory partitions of 100K, 500K, 200K, 300K, and 600K (in order), how would each of the First-fit, Best-fit, and Worst-fit algorithms place processes of 212K, 417K, 112K, and 426K (in order)? Which algorithm makes the most efficient use of memory?
2. Consider the following page reference string 7,0,2,0,3,0,4,2,3,0,3,2,1,2,0,1,7,0,2.

How many page faults would occur in the case?

a. LRU

b. FIFO

c. Optimal algorithms

assuming three, five or six frames. Note that initially all frames are empty.

3.. Assume that we have a paging system with page table stored in memory

a. If a memory reference takes 200 nanoseconds how long does a paged memory reference take? b.

If we add associative registers and 75% of all page table references are found in the associative registers, what is the effective memory reference time? Assume that finding a page table entry in the associative registers takes zero time, if the entry is there.

4. Consider a demand-paging system with the following time-measured utilizations

CPU utilization	20%
Paging disk	97.7%
Other I/O devices	5%

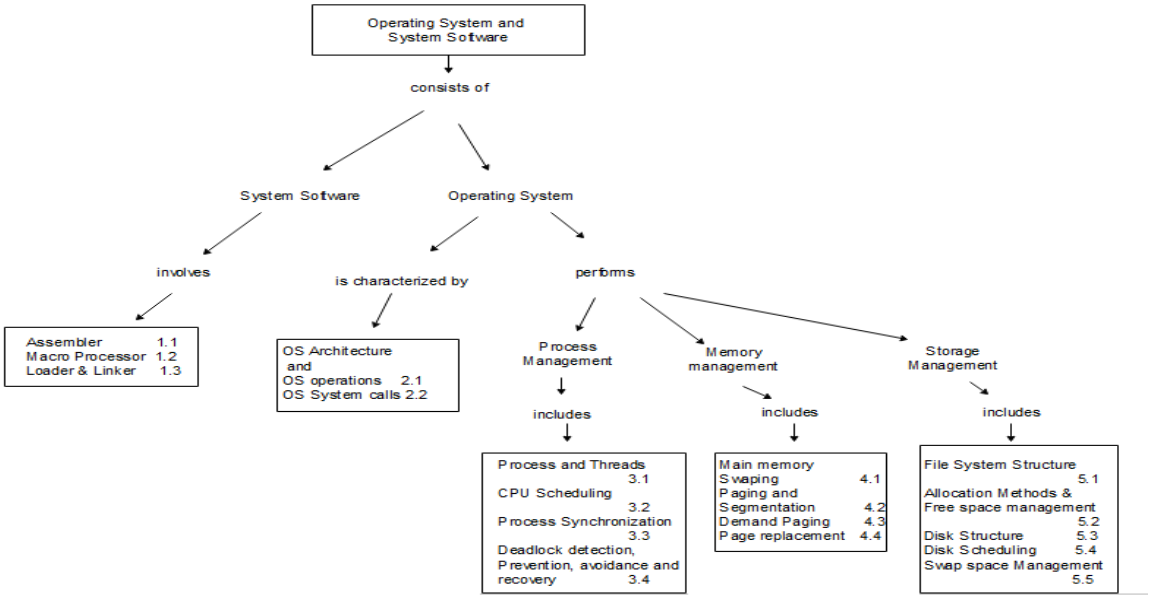
Which (if any) of the following will (probably) improve CPU utilization? Explain your answer.

a. Install a faster CPU.
b. Install a bigger paging disk.
c. Increase the degree of multiprogramming.
d. Decrease the degree of multiprogramming.
e. Install more main memory.
f. Install a faster hard disk or multiple controllers with multiple hard disks.
g. Add pre paging to the page fetch algorithms.
h. Increase the page size.

Course Outcome 6(CO6):

1. Put the following disk scheduling policies in the order that will result in minimum amount of head movement. a. FCFS b. Circular scan c. Elevator algorithm
2. Suppose that a disk drive has 5000 cylinders, numbered from 0 to 4999. the drive is currently serving a request at cylinder 143, and the previous request was at cylinder 125. The queue of pending requests, in FIFO order is 86, 1470, 913, 1774, 948, 1509, 1022, 1750, 130. starting from the current head position, what is the total distance (in cylinders) that the disk arm moves to satisfy all the pending requests for each of the disk scheduling algorithms. a. FCFS b. SSTF c. SCAN d. LOOK e. C-SCAN f. C-LOOK.
3. Explain the different type of directories in the UNIX system
4. Explain directory paths
5. Distinguish the difference between COM and EXE program

Concept Map



Syllabus

Assemblers & Macro Processors: Simple Assembly Scheme, Pass Structure of assemblers, Macro Definition and Call, Macro Expansion, Nested Macro Calls, **Linkers and Loaders:** Introduction, Relocation and linking Concepts, Types of Loaders. **Operating System Introduction & Structure:** Basics, OS Architecture, OS Operations, System calls. **Process Management:** Processes, CPU Scheduling, Process synchronization, Deadlock Detection, Deadlock Prevention, Deadlock Avoidance, Deadlock Recovery. **Memory Management:** Main Memory – swapping, Paging, Segmentation, Virtual Memory – Demand paging, Page Replacement, **Storage Management:** File System structure, Allocation methods, free space management, Disk Structure, Disk Scheduling, Swap-Space Management.

Learning Resources

1. D.M.Dhamdhere : System Programming, Tata McGraw Hill, 2011.
2. Avi Silberschatz, Peter Baer Galvin and Greg Gagne: Operating System Concepts, Eighth edition, John Wiley and Sons, 2009.
3. Leland L.Beck, System Software – An Introduction to System Programming, Pearson Education, Third Edition, 2011.
4. Andrew S. Tanenbaum, Albert S.WoodHull: Operating Systems, Design and Implementation, Third Edition, Prentice Hall, 2006.
5. William Stallings: Operating Systems: Internals and Design Principles, Fifth Edition, Prentice Hall, 2004.

Course Contents and Lecture Schedule

Module No.	Topic	No. of Hours	Course Outcome
1	**System Software**	**(6)**	
1.1	Assembler -Simple Assembly Scheme Concepts	2	CO1
1.2	Macro Processors - Macro Definition and Call Concepts	3	CO2
1.3	Linkers and Loaders - Relocation and linking Concepts	1	CO1
2	**Operating System Introduction & Structure**	**(4)**	
2.1	OS Architecture and Operations	2	CO3
2.2	OS System Calls, OS Structure	2	CO3
3	**Process Management**	**(10)**	
3.1	Processes and Threads	2	CO3
3.2	CPU Scheduling	2	CO4
3.3	Process synchronization	2	CO4
3.4	Deadlock Detection, Prevention, Avoidance, Recovery	4	CO4
4	**Memory Management**	**(9)**	
4.1	Main Memory – swapping	2	CO5
4.2	Paging and Segmentation	3	CO5
4.3	Virtual Memory – Demand paging	2	CO5
4.4	Page Replacement	2	CO5
5	**Storage Management**	**(7)**	
5.1	File System Structure	1	CO6
5.2	Allocation methods and free space management	2	CO6
5.3	Disk Structure	1	CO6
5.4	Disk Scheduling	2	CO6
5.5	Swap-Space Management	1	CO6
	Total No of Hours	**36**	

Course Designers:

1. Dr.P.Chitra pccse@tce.edu
2. Mr.K.NarasimhaMallikarjunan arjunkambaraj@tce.edu

18CS440	DATABASE MANAGEMENT SYSTEMS

Category	L	T	P	Credit
PC	2	1	0	3

Preamble

This course aims at facilitating the student to understand the various concepts and functionalities of Database Management Systems, the method and model to store data and how to manipulate the data through relational and query languages. The course also covers the effective designing of relational database and how the system manages the concurrent usage of data in a multiuser environment.

Prerequisite

Data Structures and Algorithms.

Course Outcomes

On the successful completion of the course students will be able to

CO Number	Course Outcome Statement	Weightage in %
CO1	Develop Entity Relationship (ER) and Relational models for a given application.	15
CO2	Manipulate relational database using Relational and Structured Query languages.	20
CO3	Develop a normalized database for a given application by incorporating various constraints like integrity and value constraints.	20
CO4	Demonstrate and explain how an optimized query will be build by the database system for data retrieval.	5
CO5	Illustrate different forms of transactions, concurrency control and recovery mechanisms to preserve data consistency in a multi user environment.	20
CO6	Construct data structures like indexes and hash tables for the fast retrieval of data.	20

CO Mapping with CDIO Curriculum Framework

CO #	TCE Proficiency Scale	Learning Domain Level			CDIO Curricular Components (X.Y.Z)
		Cognitive	Affective	Psychomotor	
CO1	TPS3	Apply	Value	Mechanism	1.2, 2.1.1, 2.1.2, 2.1.5, 4.4.3, 4.4.4
CO2	TPS3	Apply	Value	Mechanism	1.2, 2.1.1, 2.1.2, 2.1.5, 4.4.3, 4.4.4
CO3	TPS3	Apply	Value	Mechanism	1.2, 2.1.5, 4.4.3, 4.4.4
CO4	TPS2	Understand	Respond	Guided Response	1.2, 2.1.1
CO5	TPS3	Apply	Value	Mechanism	1.2, 2.1.1, 2.1.2, 2.1.5, 4.4.3, 4.4.4
CO6	TPS3	Apply	Value	Mechanism	1.2, 2.1.1, 2.1.5, 4.4.3, 4.4.4

Mapping with Programme Outcomes and Programme Specific Outcome

COs	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	S	M	L		M	M	L	L	M	M		M	M	L	L
CO2	S	M	L		M	M	L	L	M	M		M	M	L	L
CO3	S	M	L		M	M	L	L	M	M		M	M	L	L
CO4	M	L											L		
CO5	S	M	L		L	L	L		L	L		M	M	L	L
CO6	S	M	L			L	L		L	L		M	M	L	L

S- Strong; M-Medium; L-Low

Assessment Pattern: Cognitive Domain

Cognitive Levels	Continuous Assessment Tests			Assignment			Terminal Examination
	1	2	3	1	2	3	
Remember	10	10	10	-	-	-	10
Understand	30	40	30	-	-	-	30
Apply	60	50	60	100	100	100	60
Analyse	-	-	-	-	-	-	-
Evaluate	-	-	-	-	-	-	-
Create	-	-	-	-	-	-	-

Assessment Pattern: Psychomotor

Psychomotor Skill	Miniproject /Assignment/Practical Component
Perception	-
Set	-
Guided Response	10
Mechanism	90
Complex Overt Responses	-
Adaptation	-
Origination	-

Sample Questions for Course Outcome Assessment

Course Outcome1 (CO1):

1. What are mapping cardinalities? State their uses with examples. (Remember)
2. Explain the concepts of generalization – specialization in E-R Model with suitable examples. (Understand)
3. For the following employee database
   employee(employee-name, street, city)
   works(employee-name, company-name, salary)
   company(company-name, city)
   manages(employee-name, manager-name)
   Construct the ER and relational models with all possible cardinalities. (Apply)

Course Outcome2 (CO2):

1. Recall the use of CHECK constraint. (Remember)
2. Distinguish between relational algebra and relational calculus with suitable example. (Understand)
3. Considering the schema structure given below

   CUSTOMER(custno, custname, city, phone)
   ITEM (Itemno, Itemname, Itemprice, QtyOnhand)
   INVOICE (Invno , Invdate , Custno)
   INVITEM (Invno , Itemno , Qty)

   Answer the following queries in SQL. (Apply)
   a. Find customers from 'Chennai '.
   b. Display all item name along with the quantity sold.

Course Outcome3 (CO3):

1. Outline the desirable properties of decomposition. (Remember)
2. When a relation is said to be in 1NF? Illustrate with an example. (Understand)
3. Develop a relational database for the Banking environment by following the various design phases of normalization. (Apply)

Course Outcome4 (CO4):

1. Define the term "evaluation plan". (Remember)
2. Let relations r1(A,B,C) and r2(C,D,E) have the following properties: r1 has 20,000 tuples, r2 has 45,000 tuples, 25 tuples of r1 fit on one block, and 30 tuples of r2 fit on one block. Estimate the number of block transfers and seeks required using Hash join strategy for r1 natural joined with r2. (Understand)
3. For the query

   select T.branch_name from branch T, branch S
   where T.assests > S.assests and S.branch_city='madurai'

   write an efficient relational_algebra expression for this query. (Understand)

Course Outcome5 (CO5):

1. What is the goal of recovery systems in DBMS? (Remember)
2. When deadlock will occur in concurrent transactions? Explain with an example. (Understand)
3. Consider the following ordering Schedule - S of transactions:

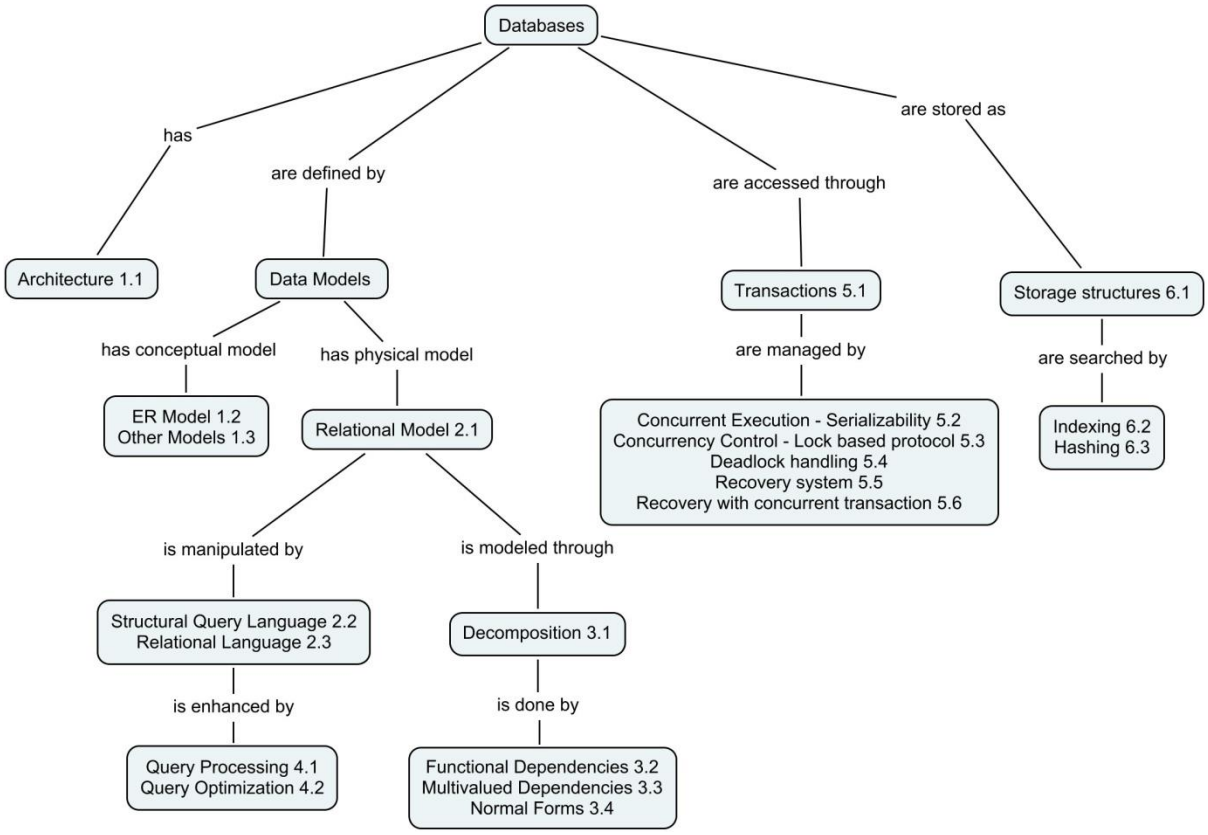
   T3: W(X); T2: R(X); T3: commit; T1: W(Y); T1: commit; T2: R(Y);
   T2: W(Z); T2: commit; T4: R(X); T4: R(Y); T4:W(Z); T4: commit.

   Identify whether given transactions are view serializable. (Apply)

Course Outcome6 (CO6):

1. When a multi-level index is required? (Remember)
2. List the advantages of dynamic hashing when compared to static hashing. (Understand)
3. Construct a B+ tree for the following set of key values { 2,3,5,7,11,17,19,23,29,31 } Suppose that we are using extendable hashing on a file that contains records with the following search key values 2,3,5,7,11,17,19,23,29,31. Show the extendable hash structure for this file, if the hash function is h(x)=x mod 8 and buckets can hold 3 records. (Apply)

Concept Map



Syllabus

Introduction to database: Purpose of database system, System Architecture, Components of DBMS, Data Models – ER model, Other models: UML model, Graph model.

Relational Databases: Relational Model – Concept of relation, Constraints, Mapping ER model to Relational Model, Structured Query Language - DDL, DML, TCL and DCL, Relational language – Relational algebra, tuple calculus.

Database design: Decomposition, Functional Dependencies - Armstrong's axioms for FD's, Closure set of FD's for key identification, Multivalved Dependencies, and Normal forms – 1NF, 2NF, 3NF, BCNF and 5NF.

Query Processing and Optimization - Measures and Evaluation of query, Estimation and Evaluation of quey, Database tuning, HBase.

Transaction and Concurrency control - Transaction concepts, Concurrent Execution, Serializability, Concurrency Control - Lock based protocol, Deadlock handling. Recovery System - Failure Classification, Backup, Log-based Recovery, Recovery with Concurrent Transaction.

Data Storage and structures: RAID levels, Database Compression, De-Duplication, File Structure - Indexing – Ordered Index - B+ tree, Hashing - Static and dynamic hashing.

Learning Resources

1. Avi Silberschatz, Henry F.Korth, S.Sudarshan: “Database System Concepts”, 6th Edition, Tata McGrawHill, 2010.
2. Sharad Maheshwari, Ruchin Jain: “Database Management System-Complete practical Approach”, Firewall Media, Second Edition, 2006.
3. Ramez Elmasri and Shamkant B.Navathe, “ Fundamentals of Database System”, 7th edition, Pearson Education, 2017.
4. Raghu Ramakrishnan, Johannes Gehrke, “Database Management Systems”, 3rd Edition, Tata McGraw Hill, 2002.
5. https://onlinecourses.nptel.ac.in/noc18_cs15/preview
6. https://courses.tce.edu/

Course Contents and Lecture Schedule

Module No.	Topic	No. of Lectures	Course Outcome
1	**Introduction to database (3)**		
1.1	Purpose of database system, System Architecture, Components of DBMS	1	CO1
1.2	Data Models – ER Model - Tutorial	1	CO1
1.3	Other Data Models - UML Model, Graph model	1	CO1
2	**Relational Databases (9)**		
2.1.1	Relational Model, Mapping ER model to Relational Model	1	CO1
2.1.2	Relational model construction - DDL – Create, Alter, Drop, Constarints - Tutorial	1	CO1
2.2.1	Structured Query Language – Introduction. DML – Set operators	1	CO2
2.2.2	Structured Query Language – DML – Joins - Tutorial	1	CO2
2.2.3	Structured Query Language – DML – Sub-queries, Nested Queries - Tutorial	1	CO2
2.2.4	Structured Query Language – DML – Complex queries, TCL and DCL	1	CO2
2.3.1	Relational language - Relational Algebra – Selection, Projection, Cross product, Join and Set operators	1	CO2
2.3.2	Relational Algebra Operations - Tutorial	1	CO2
2.3.3	Relational language - Relational calculus	1	CO2
3	**Database design (7)**		
3.1	Decomposition	1	CO3
3.2	Functional Dependencies - Armstrong's axioms for FD's, Closure set of FD's for key identification	1	CO3
3.3	Functional Dependencies, Closure set - Tutorial	1	CO3
3.4	Multivalved Dependencies	1	CO3
3.5	Normal forms - 1NF, 2NF, 3NF, BCNF and 5NF	2	CO3
	Normalization - Tutorial	1	CO3
4	**Query Processing and Optimization (3)**		
4.1	Query Processing - Measures and Evaluation of query	1	CO4
4.2.1	Query Optimization - Estimation and Evaluation of quey	1	CO4
4.2.2	Query Optimization - Database tuning, HBase	1	CO4
5	**Transaction and Concurrency control (8)**		

Module No.	Topic	No. of Lectures	Course Outcome
5.1	Transaction concepts	1	CO5
5.2	Concurrent Execution – Serializability - Tutorial	1	CO5
5.3	Concurrency Control - Lock based protocol	1	CO5
5.4	Deadlock handling - Tutorial	1	CO5
5.5	Recovery – Backup, Log based recovery	1	CO5
5.6	Tutorial on Recovery	1	CO5
5.7	Recovery with concurrent transaction	2	CO5
6	**Data Storage and stuctures (6)**		
6.1	Storage structures – RAID levels, Database Compression, De-Duplication, File Structure	2	CO6
6.2	Indexing - Ordered Index – B+ tree	1	CO6
6.3	Tutorial on B+ indexing	1	CO6
6.4	Hashing - Static and dynamic hashing	1	CO6
6.5	Tutorial on hashing	1	CO6
	Total	**36**	

Course Designers:

1. Mrs. A.M.Rajeswari amrcse@tce.edu
2. Mrs.B.Subbulakshmi bscse@tce.edu

18EG460	PROFESSIONAL COMMUNICATION

Category	L	T	P	Credit
HSS	0	1	2	2

Preamble

This course helps the students to achieve effective language proficiency for their professional, social and interpersonal communication skills, hence increasing their employability and career skills.

Prerequisite

Basic English Knowledge

Course Outcomes

On the successful completion of the course students will be able to

CO Number	Course Outcome Statement	Weightage*** in %
CO1	Listen, watch, understand and respond to talks, conversations, etc by native and neutral speakers on science, general context, and from ETS test samples with confidence.	22%
CO2	Present ideas, express opinions/comments, practice presentation, and converse in discussions on a variety of technical and non-technical domains without fear	39%
CO3	Read and comprehend passages/texts from various topics – general and reasoning, to respond precisely through reading techniques, besides getting awareness on competitive exam lexicon/verbal exercises for career prospects	17%
CO4	Write journal abstracts/projects and business correspondences with clarity, accuracy, intelligibility, and precision.	22%

*** Weightage depends on Bloom's Level, number of contact hours

CO Mapping with CDIO Curriculum Framework

CO #	TCE Proficiency Scale	Learning Domain Level			CDIO Curricular Components (X.Y.Z)
		Cognitive	Affective	Psychomotor	
CO1	TPS2	Understand	Respond	Guided Response	2.4.2, 2.4.6, 3.2.1, 3.2.2,
CO2	TPS3	Apply	Value	Mechanism	3.1.3, 3.1.2, 3.2.4, 3.2.5,3.2.6
CO3	TPS2	Understand	Respond	Guided Response	2.4.6, 2.4.5, 3.2.1,
CO4	TPS3	Apply	Value	Mechanism	2.4.3, 3.2.1, 3.2.3, 3.2.5

Mapping with Programme Outcomes and Programme Specific Outcomes

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO 1									L	S		M			L
CO 2									S	S		M			M
CO 3									M	S		M			L
CO 4									M	S		M			L

S- Strong; M-Medium; L-Low

Assessment Pattern:

Internal: No Continuous Assessment Test(CAT) will be conducted. Students' performance will be continuously assessed in various classroom activities in Listening, Speaking, Reading and Writing for 50 marks as detailed below:

Listening Test - 10
Speaking Test (Group Discussion and Technical Presentation) - 20
Written Test(Objective/Descriptive to be tested for 40 marks and converted to 20 marks)- 20

External (Practical):

Group Discussion - 20
Personal Interview / Situational Conversation (BEC speaking based) - 20
Listening Test - 20
Reading / Writing – Computerised or Paper-based Test / General Aptitude Test – Objective type - 40

List of Experiments/Activities with CO Mapping

S.No	Activities	Hours		CO Mapping			
		T	P				
1	Listening, Reading and Writing based on Extensive	2		CO1		CO3	CO4
2	Listening exercises at lab - online resources		2	CO1			
3	Developing Listening skills (BEC / IELTS / TOEIC /		2	CO1			
4	GD/Mock interview/Presentation Intro at lab through		2	CO1			
5	GD Practice at classroom in groups		4	CO1	CO2		
6	Presentation on Technical / general topics – from dailies	1	4		CO2		
7	Mock interview practice at classroom	1	4	CO1	CO2		
8	Comprehension Descriptive and Reasoning	2	2			CO3	
9	General Aptitude Practice – Vocabulary Development / Sentence completion / Error spotting /Analogy /	3	2			CO3	CO4
10	Business Correspondence - BEC Writing Task II	2					CO4
11	Basics of Technical Writing/ Project Reports		2		CO2		
12	Preparation of Resume	1					CO4

Learning Resources

Reference Books:

1. Cappel, Annette and Sharp, Wendy, Cambridge English: Objective First, 4th Ed., CUP, New Delhi, 2013.
2. Cusack, Barry. Improve Your IELTS Listening and Speaking Skills (With CD) Paperback, Mcmillan, 2007.
3. Bates, Susan TOEFL iBT Exam Paperback – Oxford, 2012.
4. Hart, Guy Brook. Cambridge English Business Benchmark: 2 Ed., CUP 2014

Websites:

1. https://ielts-up.com (IELTS – LSRW – Practice Tests)
2. www.cambridgeenglish.org (BEC - LSRW)
3. www.etsglobal.org (TOEIC Preparation)
4. www.examenglish.com (Online Exams for international ESL Exams)

5. www.testpreppractice.net (GRE Tests -Vocabulary /Analogy / Sentence Completion / Reading)
6. https://www.freshersworld.com (Placement Papers)

Extensive Reading:

Coelho, Paulo. The Alchemist, Harper Publication, 2018.

Course Designers:

1. Dr.A.Tamilselvi , Convenor
2. Dr S.Rajaram
3. Mr.Vinoth.R
4. Dr.G.Jeya Jeevakani
5. Ms.R.Manibala

18CS470	SYSTEM SOFTWARE AND OPERATING SYSTEMS LAB

Category	L	T	P	Credit
PC	0	0	2	1

Preamble

This laboratory enables the students clearly understand the concepts of system software. Also students can implement the scheduling, process and memory management techniques.

Prerequisite

Computer Organization and Microprocessor

Problem Solving using Computers

Course Outcomes

On the successful completion of the course students will be able to

CO Number	Course Outcome Statement	Weightage*** in %
CO1	Implement system software components like symbol table generator, single-pass and multi-pass assemblers (Apply)	25
CO2	Implement and demonstrate the function of system software like loaders, linkers in program execution (Apply)	8
CO3	Implement page replacement algorithms and dynamic storage allocation algorithms. utilize the first fit and best fit algorithms for allocating and managing memory (Apply)	18
CO4	Implement scheduling algorithms such as FCFS, SJFS and Round Robin to schedule a given set of processes. (Apply)	17
CO5	Implement Banker's algorithm for deadlock avoidance (Apply)	8
CO6	Implement disk scheduling algorithms like FCFS, SSTF, SCAN and C-SCAN. (Apply)	8
C07	Construct programs to demonstrate inter-process communication using shared memory, pipes and message queues. (Apply)	8
C08	Implement solutions to the critical section problem using semaphores. (Apply)	8

CO Mapping with CDIO Curriculum Framework

CO #	TCE Proficiency Scale	Learning Domain Level			CDIO Curricular Components (X.Y.Z)
		Cognitive	Affective	Psychomotor	
CO1	TPS3	Apply	Value	Mechanism	1.2, 4.5.1
CO2	TPS3	Apply	Value	Mechanism	1.2, 4.5.1
CO3	TPS3	Apply	Value	Mechanism	1.2, 4.5.1
CO4	TPS3	Apply	Value	Mechanism	1.2, 4.5.1
CO5	TPS3	Apply	Value	Mechanism	1.2, 4.5.1
CO6	TPS3	Apply	Value	Mechanism	1.2, 4.5.1
CO7	TPS3	Apply	Value	Mechanism	1.2, 4.5.1
CO8	TPS3	Apply	Value	Mechanism	1.2, 4.5.1

Mapping with Programme Outcomes and Programme Specific Outcomes

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	S	M	L		L			S	L	M		M	M		M
CO2	S	M	L		L			S	L	M		M	M		M

CO3	S	M	L		L			S	L	M		M	M		M
CO4	S	M	L		L			S	L	M		M	M		M
CO5	S	M	L		L			S	L	M		M	M		M
CO6	S	M	L		L			S	L	M		M	M		M
CO7	S	M	L		L			S	L	M		M	M		M
CO8	S	M	L		L			S	L	M		M	M		M

S- Strong; M-Medium; L-Low

Assessment Pattern: Cognitive Domain

Cognitive Levels	Model Examination	Terminal Examination
Remember		
Understand		
Apply	100	100
Analyse		
Evaluate		
Create		

Assessment Pattern: Psychomotor

Psychomotor Skill	Miniproject /Practical Component/Observation
Perception	
Set	
Guided Response	15
Mechanism	85
Complex Overt Responses	
Adaptation	
Origination	

List of Experiments/Activities with CO Mapping

Sl.No	Experiments	CO
1.	Implementation of symbol table.	CO1
2.	Implementation of Single pass assembler.	CO1
3.	Implementations of Multi pass assembler.	CO1
4.	Implementation of Linker, absolute and relocatable Loader.	CO2
5.	Given the list of processes, their CPU burst times and arrival times, display/print the Gantt chart for FCFS and SJF. For each of the scheduling policies, compute and print the average waiting time and average turnaround time.	CO4
6.	Given the list of processes, their CPU burst times and arrival times, display/print the Gantt chart for Priority and Round robin. For each of the scheduling policies, compute and print the average waiting time and average turnaround time.	CO4
7.	Developing Application using Inter Process Communication (using shared memory, pipes or message queues)	CO7

8.	Implement Bankers Algorithm for Deadlock Avoidance	CO5
9.	Implement the Producer – Consumer problem using semaphores (using UNIX system calls).	CO8
10.	Implement First fit and Best fit memory management schemes	CO3
11.	Implement First come first serve and LRU Page replacement Algorithms	CO3
12.	Implement Disk management using Algorithms such as FCFS,SSTF,SCAN and C-SCAN	CO6

Learning Resources

1. Avi Silberschatz, Peter Baer Galvin and Greg Gagne: Operating System Concepts, Seventh edition, John Wiley and Sons, 2006.

Course Designers

1. Dr. P.Chitra pccse@tce.edu

2. Mr. K.Narasimha Mallikarjunan arjunkambaraj@tce.edu

18CS480	**ALGORITHMS LAB**

Category	L	T	P	Credit
PC	0	0	2	1

Preamble

The objective of this laboratory course is to enable students to solve algorithmic problems by choosing and/or designing efficient data structures and algorithms to meet the problem constraints and implementing the algorithm in C/C++ and Python.

Prerequisite

Problem Solving using Computers

Data Structures and Algorithms

Data Structures Lab

Course Outcomes

On the successful completion of the course students will be able to

CO Number	Course Outcome Statement	Weightage*** in %
CO1	Determine the time complexity of various sorting and searching techniques.	15
CO2	Apply design techniques like divide and conquer to solve the recursive problems.	15
CO3	Identify the problem and design the algorithm using greedy dynamic programming techniques.	15
CO4	Implement various graph traversal algorithm and analyse their performance.	15
CO5	Develop efficient algorithms using branch and bound techniques for solving real world, scientific and engineering problems.	15
CO6	Design, implement and evaluate the algorithms designed using a high-level programming language.	25

CO Mapping with CDIO Curriculum Framework

CO #	TCE Proficiency Scale	Learning Domain Level			CDIO Curricular Components (X.Y.Z)
		Cognitive	Affective	Psychomotor	
CO1	TPS3	Apply	Value	Mechanism	1.2,2.1.1,2.4.3
CO2	TPS3	Apply	Value	Mechanism	1.2,2.1.1,2.4.3
CO3	TPS4	Analyse	Organise	Complex Overt Responses	1.2,2.1.1,2.4.3
CO4	TPS3	Apply	Value	Mechanism	1.2,2.1.1,2.4.3
CO5	TPS3	Apply	Value	Mechanism	1.2,2.1.1,2.4.3
CO6	TPS5	Evaluate	Organise	Adaptation	1.2,2.1.1,3.1.1,4.5.1

Mapping with Programme Outcomes and Programme Specific Outcomes

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	S	M	L					M	M	M		M	M		M
CO2	S	M	L					M	M	M		M	M		M
CO3	S	S	M	L				M	M	M		M	S		M
CO4	S	M	L					M	M	M		M	M		M
CO5	S	M	L					M	M	M		M	M		M
CO6	S	S	S	M				M	M	M		M	S		M

S- Strong; M-Medium; L-Low

Assessment Pattern: Cognitive Domain

Cognitive Levels	Model Examination	Terminal Examination
Remember		
Understand		
Apply	20	30
Analyse	20	30
Evaluate	10	20
Create		

Assessment Pattern: Psychomotor

Psychomotor Skill	Miniproject /Practical Component/Observation
Perception	
Set	
Guided Response	
Mechanism	
Complex Overt Responses	60
Adaptation	40
Origination	

List of Experiments/Activities with CO Mapping

Students will be expected to understand and model the problem, design efficient algorithms, implement the algorithms designed using C/C++

1. Analysis of searching (Linear/Binary search)and sorting algorithm(O(n^2)/O(nlogn)
2. Analysis of divide & conquer algorithms – Merge sort/Quick sort
3. Implementation of greedy algorithm- Prim's / Kruskal /Knapsack problem
4. Implementation of dynamic programming – matrix multiplication
5. Application of DFS – Topological sort/Strongly connected components
6. Implementation of maximum flow algorithm – Ford Fulkerson algorithm
7. Implementation of branch and bound techniques – Travelling salesman problem
8. Mini Project - Identifying an appropriate data structure and implement it using relevant algorithms and analyse the time complexity to solve problems like :
    - Huffman coding and decoding
    - Maximum flow in a dynamic network
    - Graph colouring
    - Drunken Donuts, a new wine-and-donuts restaurant chain, wants to build restaurants on many street corners with the goal of maximizing their total profit. The street network is described as an undirected graph G = (V, E), where the potential restaurant sites are the vertices of the graph. Each vertex u has a nonnegative integer value pu, which describes the potential profit of site u. Two restaurants cannot be built on adjacent vertices (to avoid selfcompetition). You are supposed to design an algorithm that outputs the chosen set $U \subseteq V$ of sites that maximizes the total profit $u \in U$ pu.

Module No.	Topic	No. of Lectures	Course Outcome
1	Analysis of searching (Linear/Binary search)and sorting algorithm(O(n^2)/O(nlogn)	3	CO1
2	Analysis of divide & conquer algorithms – Merge sort/Quick sort	3	CO2
3	Implementation of greedy algorithm- Prim's / Kruskal /Knapsack problem	3	CO3
4	Implementation of dynamic programming – matrix multiplication	3	CO3
5	Application of DFS – Topological sort/Strongly connected components	3	CO4
6	Implementation of maximum flow algorithm – Ford Fulkerson algorithm	3	CO4
7	Implementation of branch and bound techniques – Travelling salesman problem	3	CO5
8	Mini Project	4	CO6
	Total	**24**	

Learning Resources

1. https://www.geeksforgeeks.org/fundamentals-of-algorithms/
2. https://www.hackerrank.com/domains/algorithms
3. https://www.codechef.com/wiki/tutorials
4. Steven S. Skiena, The Algorithm Design Manual, Second Edition, Springer,2010.

19CS490	PROJECT MANAGEMENT

Category	L	T	P	Credit
HSS	3	0	0	3

Preamble

This course develops the competencies and skills for planning, organizing and controlling projects and understanding interpersonal issues that leads to successful project outcomes. Discusses the activities for the prospective project managers to manage their teams, schedules, risks and resources to produce the desired outcome. The course is based on references from Project Management Institute's (PMI®) Project Management Body of Knowledge (PMBOK® Guide).

Prerequisite

Nil

Course Outcomes

On the successful completion of the course students will be able to

CO Number	Course Outcome Statement	Weightage*** in %
CO1	Suggest an organizational structure for managing projects to develop a product with a given specification	10
CO2	Develop a project communication plan by defining its scope, priorities and responsibility matrices	20
CO3	Construct a work breakdown structure for a given business cases	20
CO4	Identify the critical path in scheduling a set of project-activities by using the Activity-On-Node method.	20
CO5	Outline the importance and various activities performed for resource management, risk assessment and project closure.	15
CO6	Plan and implement a team-project for developing a product.	20

*** Weightage depends on Bloom's Level, number of contact hours,

CO Mapping with CDIO Curriculum Framework

CO #	TCE Proficiency Scale	Learning Domain Level			CDIO Curricular Components (X.Y.Z)
		Cognitive	Affective	Psychomotor	
CO1	TPS3	Apply	Value	Mechanism	1.2,2.5
CO2	TPS3	Apply	Value	Mechanism	1.2,2.5
CO3	TPS3	Apply	Value	Mechanism	1.2,2.2.1,2.2.3
CO4	TPS3	Apply	Value	Mechanism	1.2,2.2.1,2.2.3,3.1.1
CO5	TPS2	Understand	Respond	Guided Response	1.2,4.1.1, 4.1.2
CO6	TPS3	Apply	Value	Mechanism	1.2, 2.2.1,3.1.1,3.1.2,3.1.4, 4.4.2

Mapping with Programme Outcomes and Programme Specific Outcomes

Cos	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	S	M	L						M	M	M		M		M
CO2	S	M	L		S				M	M	M		M	L	M
CO3	S	M	L						M	M	M		M		M

CO4	S	M	L		S				M	M	M		M	L	M
CO5	M	L											M		
CO6	S	M	L		S			M	M	M	M	M	M	L	M

S- Strong; M-Medium; L-Low

Assessment Pattern: Cognitive Domain

Cognitive Levels	Continuous Assessment Tests			Assignment			Terminal Examination
	1	2	3	1	2	3	
Remember	20	20	20	-	-	-	20
Understand	40	40	40	-	-	-	50
Apply	40	40	40	100	100	100	30
Analyse							
Evaluate							
Create							

Assessment Pattern: Psychomotor

Psychomotor Skill	Mini project /Assignment/Practical Component
Perception	
Set	
Guided Response	30
Mechanism	70
Complex Overt Responses	
Adaptation	
Origination	

Sample Questions for Course Outcome Assessment**

Course Outcome 1 (CO1):

1. Suggest a suitable organisational structure for a Central Engineering Systems, Inc.
2. You work for LL Company, which manufactures high-end optical scopes for hunting rifles. LL Company has been the market leader for the past 20 years and has decided to diversify by applying its technology to develop a top-quality binocular. What kind of project management structure would you recommend they use for this project? What information would you like to have to make this recommendation, and why?
3. Explain how organizing projects are deployed within the dedicated project teams. State its pros and cons.

Course Outcome 2 (CO2):

1. Estimate the cost associated with the project using bottom up technique
2. Why is the implementation of projects important to strategic planning and the project Manager ?
3. Explain the guidelines for estimating times, costs and resources and explain the methods for estimating the project costs by Top Down Approach

Course Outcome 3 (CO3):

1. Develop a work breakdown structure for a wedding

2 How does the WBS differ from the project network
3. Develop a WBS for a project in which you are going to build a bicycle. Try to identify all of the major components and provide three levels of detail.

Course Outcome 4 (CO4):

1. Draw a project network from the following information. What activity(s) is a burst activity? What activity(s) is a merge activity?

ID	Description	Predecessor	Time
A	Survey	None	5
B	Soils report	A	20
C	Traffic design	A	30
D	Lot layout	A	5
E	Approve design	B,C,D	80
F	Illumination	E	15
G	Drainage	E	30
H	Landscape	E	25
I	Signing	E	20
J	Bid proposal	F,G,H,I	10

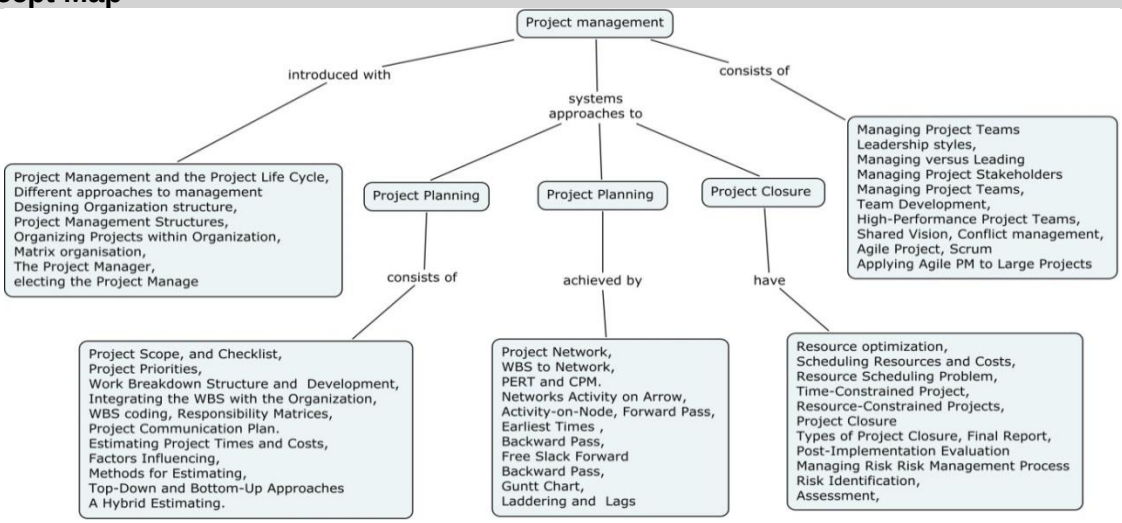
2. Create a customer database for the Modesto league baseball team. Draw a project network Complete the forward and backward pass, compute activity slack, and identify the critical path. How long will this project take? How sensitive is the network schedule? Calculate the free slack and total slack for all noncritical activities.
3. Your roommate is about to submit a scope statement for a spring concert sponsored by the entertainment council at Western Evergreen State University (WESU). WESU is a residential university with over 22,000 students. This will be the first time in six years since WESU sponsored a spring concert. The entertainment council has budgeted $40,000 for the project. The event is to occur on June 5th. Since your roommate knows you are taking a class on project management she has asked you to review her scope statement and make suggestions for improvement. She considers the concert a resume-building experience and wants to be as professional as possible. Below is a draft of her scope statement. What suggestions would you make and why?

Course Outcome 5 (CO5):

1 What is the difference between avoiding a risk and accepting a risk?
2 What is the difference between mitigating a risk and contingency planning?
3 Develop a risk breakdown structure and describe the process involved in Risk management.

*Note: **CO6** will be attained through Mini Projects / Assignments.

Concept Map



Syllabus

Modern Project Management

Project Management and its importance, The Project Life Cycle, Different approaches to management, Organization structure and Project Management, Organizing Projects within the Functional Organization, Matrix organisation and Projectized organisation, The Project Manager, and Project Management

Project Initiation

Project Scope, and Checklist, Project Priorities, Software Project Planning, Work Breakdown Structure Development, WBS coding, Responsibility Matrices in WBS, Project Communication Plan, Project Estimation and Methods for Estimation, Software project estimation.

Project Planning

Selection of appropriate project management approaches, spiral model, Agile, Extreme programming, Iterative planning, WBS to Project Network, Networks Activity on Arrow, Activity-on-Node, Forward Pass, Earliest Times , Backward Pass—Latest Times, Laddering and Lags,Gantt Chart

Resource optimization, Scheduling Resources and Costs, Resource Scheduling Problem, Time-Constrained Project, Resource-Constrained Projects. **Managing Risk,** Risk Management Process, Risk Identification, Assessment, and Response Development Managing people in software projects

Leadership, Leadership styles, managing versus Leading a Project, Managing Project Stakeholders **Managing Project Teams**, Team Development, High-Performance Project Teams, Shared Vision, Conflict management, Project Closure Types of Project Closure, Final Report, Post-Implementation Evaluation

Learning Resources

1. By Bob Hughes, Mike Cotterell, “Software Project Management”, Tata McGraw-Hill Education.
2. Erik W. Larson, Clifford F. Gray, “Project Management The Managerial Process”, McGraw-Hill/Irwin, Fifth Edition, 2011.

3. Jack R. Meredith, Samuel J. Mantel, Jr., “Project management A Managerial Approach”., John Wiley & Sons, Inc. Seventh Edition,2009
4. Harold kerzner , “Project Management A systems approach to Planning, scheduling, And controlling”, Tenth edition, John Wiley & Sons, Inc.2009
5. Harold kerzner , “Project management best practices achieving global excellence”, Second edition, John Wiley & Sons, Inc.2010
6. A Guide to the Project Management Body of Knowledge (PMBOK® Guide)—Fifth Edition, Project Management Institute.
7. Harold Koontz, Heinz weihrich ”Essentials of Management”, Tata McGraw-Hill Education, 2006 - Management

Course Contents and Lecture Schedule

Module No.	Topic	No. of Lectures	Course Outcome
1	**Modern Project Management**		
1.1	Project Management and its importance	1	CO1
1.2	The Project Life Cycle	1	CO1
1.3	Different approaches to management,	1	CO1
1.4	Organization structure and Project Management	2	CO1
1.5	Organizing Projects within the Functional Organization, Matrix organisation and Projectized organisation	1	CO1
1.6	The Project Manager, and Project Management	1	CO1
2	**Project Initiation**		
2.1	Project Scope, and Checklist, Project Priorities,	1	CO2
2.2	Software Project Planning	2	CO2
2.3	Work Breakdown Structure Development, WBS coding.	1	CO3
2.4	Responsibility Matrices in WBS, Project Communication Plan.	2	CO2
2.5	Project Estimation and Methods for Estimation Software project estimation	2	CO2
3.1	**Project Planning**		
3.2	Selection of appropriate project management approaches, spiral model,	2	CO4
3.3	Agile, Extreme programming, Iterative planning	1	CO4
3.4	WBS to Project Network	2	CO4
3.5	Networks Activity on Arrow, Activity-on-Node	2	CO4
3.6	Forward Pass, Earliest Times , Backward Pass— Latest Times, Laddering and Lags, Gantt Chart	2	CO4
4	**Resource optimization**,		
4.1	Scheduling Resources and Costs, Resource Scheduling Problem,	2	CO5
4.2	Time-Constrained Project,	1	CO5
4.3	Resource-Constrained Projects,	1	CO5
4.4	Managing Risk, Risk Management Process Risk Identification, Assessment, Response Development,	1	CO5

Module No.	Topic	No. of Lectures	Course Outcome
	Managing people in software projects		
5	**Software project management**		
5.1	Leadership styles, Managing versus Leading a Project, Managing Project Stakeholders	1	CO6
5.2	Managing Project Teams, Team Development, High-Performance Project Teams, Shared Vision	2	CO6
5.3	Conflict management.	1	CO5
5.4	Project Closure Types of Project Closure, Final Report, Post-Implementation Evaluation	2	CO5
	Total	36	

Course Designers:

1. Dr.N.Shivakumar shiva@tce.edu
2. Mr.V.Vignaraj Ananth vignaraj@tce.edu

18CHAB0	CONSTITUTION OF INDIA

Category	L	T	P	Credit
AC	2	0	0	0

Preamble

On the successful completion of the course, the students will be able to explain the basic features and fundamental principles of Constitution of India. The Constitution of India is the supreme law of India. Parliament of India cannot make any law which violates the Fundamental Rights enumerated under the Part III of the Constitution. The Parliament of India has been empowered to amend the Constitution under Article 368, however, it cannot use this power to change the "basic structure" of the constitution, which has been ruled and explained by the Supreme Court of India in its historical judgments. The Constitution of India reflects the idea of "Constitutionalism" – a modern and progressive concept historically developed by the thinkers of "liberalism" – an ideology which has been recognized as one of the most popular political ideology and result of historical struggles against arbitrary use of sovereign power by state. The historic revolutions in France, England, America and particularly European Renaissance and Reformation movement have resulted into progressive legal reforms in the form of "constitutionalism" in many countries. The Constitution of India was made by borrowing models and principles from many countries including United Kingdom and America.

The Constitution of India is not only a legal document but it also reflects social, political and economic perspectives of the Indian Society. It reflects India's legacy of "diversity". It has been said that Indian constitution reflects ideals of its freedom movement; however, few critics have argued that it does not truly incorporate our own AICTE Model Curriculum for Mandatory Courses & Activities (Non-Credit) for Undergraduate Degree in Engineering & Technology ancient legal heritage and cultural values. No law can be "static" and therefore the Constitution of India has also been amended more than one hundred times. These amendments reflect political, social and economic developments since the year 1950. The Indian judiciary and particularly the Supreme Court of India has played an historic role as the guardian of people. It has been protecting not only basic ideals of the Constitution but also strengthened the same through progressive interpretations of the text of the Constitution. The judicial activism of the Supreme Court of India and its historic contributions has been recognized throughout the world and it gradually made it "as one of the strongest court in the world"

Course Outcome:

On the successful completion of the course students will be able to

CO1	Explain the meaning of the constitution law and constitutionalism and Historical perspective of the Constitution of India	Understand
CO2	Explain the salient features and characteristics of the Constitution of India, scheme of the fundamental rights and the scheme of the Fundamental Duties and its legal status	Understand
CO3	Explain the Directive Principles of State Policy, Federal structure and distribution of legislative and financial powers between the Union and the States, and Parliamentary Form of Government in India	Understand
CO4	Explain the amendment of the Constitutional Powers and Procedure, the historical perspectives of the constitutional amendments in India, and Emergency Provisions.	Understand
CO5	Explain the Local Self Government – Constitutional Scheme in India, Scheme of the Fundamental Right to Equality,	Understand

CO6	Explain the scheme of the Fundamental Right to certain Freedom under Article 19, and Scope of the Right to Life and Personal Liberty under Article 21	Understand

Mapping with Programme Outcomes

Cos	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2	PSO 3
CO 1	M	L	-	-	-	M	-	M	-	L	-	-	L	L	L
CO 2	M	L	-	-	-	M	-	M	-	L	-	-	L	L	L
CO 3	M	L	-	-	-	M	-	M	-	L	-	-	L	L	L
CO 4	M	L	-	-	-	M	-	M	-	L	-	-	L	L	L
CO 5	M	L	-	-	-	M	-	M	-	L	-	-	L	L	L
CO 6	M	L	-	-	-	M	-	M	-	L	-	-	L	L	L

S- Strong; M-Medium; L-Low

Syllabus

1. Meaning of the constitution law and constitutionalism
2. Historical perspective of the Constitution of India
3. Salient features and characteristics of the Constitution of India
4. Scheme of the fundamental rights
5. The scheme of the Fundamental Duties and its legal status
6. The Directive Principles of State Policy – Its importance and implementation
7. Federal structure and distribution of legislative and financial powers between the Union and the States
8. Parliamentary Form of Government in India – The constitution powers and status of the President of India
9. Amendment of the Constitutional Powers and Procedure
10. The historical perspectives of the constitutional amendments in India
11. Emergency Provisions : National Emergency, President Rule, Financial Emergency
12. Local Self Government – Constitutional Scheme in India
13. Scheme of the Fundamental Right to Equality
14. Scheme of the Fundamental Right to certain Freedom under Article 19
15. Scope of the Right to Life and Personal Liberty under Article 21

Assessment Pattern

Bloom's category	Continuous Assessment Tests		Seminar
	1	2	-
Remember	40	40	0
Understand	60	60	100
Apply	0	0	0
Analyze	0	0	0
Evaluate	0	0	0

Create	0	0	0

References

1. Durga Das Basu, 'Introduction to The Constitution of India', LexisNexis Butterworths Wadhwa, 20th Edition, Reprint 2011.
2. Constitution of India, National Portal of India, Web link: https://www.india.gov.in/my-government/constitution-india

Course Designers:

1. Adapted from AICTE Model Curriculum for Undergraduate Degree Courses in Engineering & Technology, Volume-II, January 2018.