RAJIV GANDHI PROUDYOGIKI VISHWAVIDYALAYA, BHOPAL

New Scheme Based On AICTE Flexible Curricula

IV Semester Bachelor of Technology (B.Tech.)-Computer Science and Business System (CSBS)

CB-401 Operating Systems

Introduction: Concept of Operating Systems (OS), Generations of OS, Typesof OS, OS Services, Interrupt handling andSystem Calls, Basic architectural concepts of an OS, Concept of Virtual Machine, Resource Manager view, process view and hierarchical view of an OS.

Processes: Definition, Process Relationship, Different states of a Process, Process Statetransitions, Process Control Block (PCB), Context switching.

Thread: Definition, Various states, Benefits of threads, Types of threads, Concept ofmultithreads.

Process Scheduling: Foundation and Scheduling objectives, Types of Schedulers, Schedulingcriteria: CPU utilization, Throughput, Turnaround Time, Waiting Time, Response Time.

Scheduling algorithms: Pre-emptive and non-pre-emptive, FCFS, SJF, RR; Multiprocessorscheduling; Real Time scheduling: RM and EDF.

I/O Hardware: I/O devices, Device controllers, Direct Memory Access, Principles of I/O.

File Management: Concept of File, Access methods, File types, File operation, Directorystructure, File System structure, Allocation methods (contiguous, linked, indexed),Free-space management (bit vector, linked list, grouping), directory implementation(linear list, hash table), efficiency and performance.

Disk Management: Disk structure, Disk scheduling - FCFS, SSTF, SCAN, C-SCAN, Diskreliability, Disk formatting, Boot-block, Bad blocks.

Inter-process Communication: Concurrent processes, precedence graphs,Critical Section, Race Conditions, Mutual Exclusion,Hardware Solution, Semaphores, Strict Alternation, Peterson's Solution, The Producer / Consumer Problem, Event Counters, Monitors, Message Passing,Classical IPC Problems: Reader's & Writer Problem, Dinning Philosopher Problem, Barber's shop problem.

Deadlocks: Definition, Necessary and sufficient conditions for Deadlock, DeadlockPrevention, Deadlock Avoidance: Banker's algorithm, Deadlock detection and Recovery.

Concurrent Programming: Critical region, conditional critical region, monitors, concurrent languages, communicating sequential process (CSP)

Memory Management: Basic concept, Logical and Physical address maps, Memoryallocation: Contiguous Memory allocation – Fixed and variable partition–Internal and External fragmentation and Compaction.

Virtual Memory: Basics of Virtual Memory – Hardware and control structures – Localityof reference, Page allocation, Partitioning, Paging, Page fault, Working Set, Segmentation, Demand paging, PageReplacement algorithms: Optimal, First in First Out (FIFO), Second Chance (SC), Notrecently used (NRU) and Least Recently used (LRU).

Case study: UNIX/Linux OS file system, shell, filters, shell programming, programming with the standard I/O, UNIX/Linux system calls.

Laboratory

1. Unix/Linux commands (files directory, data manipulation, network communication etc), shell programming and vi editor
2. C programsfor implementation of the following:
    a. Scheduling Algorithms
    b. Shared memory
    c. Thread and Multi Thread
    d. Inter Process Communication
    e. Deadlock Avoidance and Deadlock Detection
    f. Semaphore
    g. Memory Management
    h. Indexing and Hashing
3. C Programs for implementing certain commands and a shell like Unix/Linux system shell, using the Unix/Linux System calls.

Text Books:

1. *Operating System Concepts Essentials.* Abraham Silberschatz, Peter Baer Galvin and Greg Gagne.

Reference Books:

1. *Operating Systems: Internals and Design Principles*. William Stallings.
2. *Operating System: A Design-oriented Approach*. Charles Patrick Crowley.
3. *Operating Systems: A Modern Perspective*. Gary J. Nutt.
4. *Design of the Unix Operating Systems*. Maurice J. Bach.
5. *Understanding the Linux Kernel*, Daniel Pierre Bovet, Marco Cesati.

RAJIV GANDHI PROUDYOGIKI VISHWAVIDYALAYA, BHOPAL

New Scheme Based On AICTE Flexible Curricula

IV Semester Bachelor of Technology (B.Tech.)-Computer Science and Business System (CSBS)

CB-402 Design And Analysis of Algorithms

Introduction: Characteristics of Algorithm. Analysis of Algorithm: Asymptotic analysis ofComplexity Bounds – Best, Average and Worst-Case behavior; Performance Measurements ofAlgorithm, Time and Space Trade-Offs, Analysis of Recursive Algorithms through Recurrence Relations: Substitution Method, Recursion Tree Method and Masters' Theorem.

Fundamental Algorithmic Strategies: Brute-Force, Heuristics, Greedy, Dynamic Programming, Branch andBound and Backtracking methodologies; Illustrationsof these techniques for Problem-Solving, Bin Packing, Knapsack, Travelling Salesman Problem.

Graph and Tree Algorithms: Traversal algorithms: Depth First Search (DFS) and BreadthFirst Search (BFS); Shortest path algorithms, Transitive closure, Minimum Spanning Tree,Topological sorting, Network Flow Algorithm.

Tractable and Intractable Problems: Computability of Algorithms, Computability classes – P,NP, NP-complete and NP-hard. Cook's theorem, Standard NP-complete problems andReduction techniques.

Advanced Topics: Approximation algorithms, Randomized algorithms, Class of problems beyond NP – P SPACE, Introduction to Quantum Algorithms.

Lab

Implementation of Different Algorithms based on various algorithmic strategies using C/C++

Books:

1. *Fundamental of Computer Algorithms*, E. Horowitz and S. Sahni.
2. *The Design and Analysis of Computer Algorithms*, A. Aho, J. Hopcroft and J. Ullman.

Reference Books:

1. *Introduction to Algorithms*, T. H. Cormen, C. E. Leiserson and R. L. Rivest.
2. *Computer Algorithms: Introduction to Design and Analysis*, S. Baase.
3. *The Art of Computer Programming, Vol. 1, Vol. 2 and Vol. 3*, .D. E. Knuth.

Quantum Computation and Quantum Information, Michael A. Nielsen and Isaac L. Chuang.

RAJIV GANDHI PROUDYOGIKI VISHWAVIDYALAYA, BHOPAL

New Scheme Based On AICTE Flexible Curricula

IV Semester Bachelor of Technology (B.Tech.)-Computer Science and Business System (CSBS)

CB-403 Software Engineering

Introduction: Programming in the small vs. programming in the large; software project failures and importance ofsoftware quality and timely availability; ofsoftware engineering towards successful execution of large software projects; emergence of softwareengineering as a discipline, Software Engineering Historical Development from Jackson Structured Programming to Agile Development.

Software Project Management: Basic concepts of life cycle models – different models and milestones; software project planning –identification of activities and resources; concepts of feasibility study; techniques for estimation of schedule and effort; software cost estimation models and concepts of software engineering economics; techniques of software project control and reporting; introduction to measurement of software size; introduction to the concepts of risk and its mitigation; configuration management.

Software Quality Management and Reliability: Software quality; Garvin's quality dimensions, McCall's quality factor, ISO 9126 quality factor; Software Quality Dilemma; Introduction to Capability Maturity Models (CMM and CMMI); Introduction to software reliability, reliability models and estimation.

Software Requirements Analysis, Design and Construction: Introduction to Software Requirements Specifications (SRS) and requirement elicitation techniques; techniques for requirement modelling – decision tables, event tables, state transition tables, Petri nets; requirements documentation through use cases; introduction to UML, introduction to software metrics and metrics-based control methods; measures of code and design quality.

Object Oriented Analysis, Design and Construction: Concepts -- the principles of abstraction, modularity, specification, encapsulation and information hiding; concepts of abstract data type; Class Responsibility Collaborator (CRC) model; quality of design; design measurements; concepts of design patterns; Refactoring; object-oriented construction principles; object oriented metrics.

Software Testing: Introduction to faults and failures; basic testing concepts; concepts of verification and validation; black box and white box tests; white box test coverage – code coverage, condition coverage, branch coverage; basic concepts of black-box tests – equivalence classes, boundary value tests, usage of state tables; testing use cases; transaction based testing; testing for non-functional requirements – volume, performance and efficiency; concepts of inspection; Unit Testing, Integration Testing, System Testing and Acceptance Testing.

Agile Software Engineering: Concepts of Agile Methods, Extreme Programming; Agile Process Model - Scrum, Feature; Scenarios and Stories

Laboratory

Development of requirements specification, function-oriented design using SA/SD, object-oriented design using UML, test case design, implementation using C++ and testing. Use of appropriate CASE tools and other tools such as configuration management tools, program analysis tools in the software life cycle.

Text Books:

1. *Software Engineering,* Ian Sommerville
2. *Software Engineering A Practitioner's Approach,* Roggers S. Pressman and Bruce R. Maxim.

Reference Books:

1. *The Essentials of Modern Software Engineering: Free the Practices from the Method Prisons!,* Ivar Jacobson, Harold "Bud" Lawson, Pan-Wei Ng, Paul E. McMahon and Michael Goedicke.
2. *Fundamentals of Software Engineering,* Carlo Ghezzi, Jazayeri Mehdi and Mandrioli Dino.
3. *Software Requirements and Specification: A Lexicon of Practice, Principles and Prejudices,* Michael Jackson.
4. *The Unified Development Process,* Ivar Jacobson, Grady Booch and James Rumbaugh.
5. *Design Patterns: Elements of Object-Oriented Reusable Software*, Erich Gamma, Richard Helm, Ralph Johnson and John Vlissides.
6. *Software Metrics: A Rigorous and Practical Approach*, Norman E Fenton and Shari Lawrence Pfleeger.
7. *Software Engineering: Theory and Practice*, Shari Lawrence Pfleeger and Joanne M. Atlee.
8. *Object-Oriented Software Construction*, Bertrand Meyer.
9. *Object Oriented Software Engineering: A Use Case Driven Approach* --Ivar Jacobson.
10. *Touch of Class: Learning to Program Well with Objects and Contracts* --Bertrand Meyer.
11. *UML Distilled: A Brief Guide to the Standard Object Modeling Language* --Martin Fowler.
12. *Introduction to Business Domains for Software Engineers*, Manoj Kumar Lal
13. *Knowledge Driven Development – Bridging Waterfall and Agile Methodologies* –- Manoj Kumar Lal

RAJIV GANDHI PROUDYOGIKI VISHWAVIDYALAYA, BHOPAL

New Scheme Based On AICTE Flexible Curricula

IV Semester Bachelor of Technology (B.Tech.)-Computer Science and Business System (CSBS)

CB-404 Introduction to Innovation, IP Management & Entrepreneurship

Course Outcome(s):

The major emphasis of the course will be on creating a learning system through which management students can enhance their innovation and creative thinking skills, acquaint themselves with the special challenges of starting new ventures and use IPR as an effective tool to protect their innovations and intangible assets from exploitation.

As a part of this course, students will:

- Learn to be familiar with creative and innovative thinking styles
- Learn to investigate, understand and internalize the process of founding a startup
- Learn to manage various types of IPR to protect competitive advantage

Topics to Be Covered:

UNIT – I

Innovation: What and Why?

Innovation as a core business process, Sources of innovation, Knowledge push vs. need pull innovations.

Class Discussion- Is innovation manageable or just a random gambling activity?

UNIT – II

Building an Innovative Organization

Creating new products and services, Exploiting open innovation and collaboration, Use of innovation for starting a new venture

Class Discussion- Innovation: Co-operating across networks vs. ‘go-it-alone’ approach

UNIT – III

Entrepreneurship:

- Opportunity recognition and entry strategies
- Entrepreneurship as a Style of Management
- Maintaining Competitive Advantage- Use of IPR to protect Innovation

UNIT – IV

Entrepreneurship- Financial Planning:

- Financial Projections and Valuation
- Stages of financing

- Debt, Venture Capital and other forms of Financing

UNIT – V

Intellectual Property Rights (IPR)

- Introduction and the economics behind development of IPR: Business Perspective
- IPR in India – Genesis and Development
- International Context
- Concept of IP Management, Use in marketing

UNIT – VI

Types of Intellectual Property

- Patent- Procedure, Licensing and Assignment, Infringement and Penalty
- Trademark- Use in marketing, example of trademarks- Domain name
- Geographical Indications- What is GI, Why protect them?
- Copyright- What is copyright
- Industrial Designs- What is design? How to protect?

Class Discussion- Major Court battles regarding violation of patents between corporate companies

Home Assignment:

Case study materials book will be given to students. Students are required to meet in groups before coming to class and prepare on the case for the day. Instructor may ask the student groups to present their analysis and findings to the class.

Further, the topic for class discussion will be mentioned beforehand and students should be ready to discuss these topics (in groups) in class. Students are required to meet in groups before coming to class and prepare on the topic. Few topics are mentioned below as examples. Instructor can add or change any topic as per requirement.

Topic 1- Is innovation manageable or just a random gambling activity?

Topic 2- Innovation: Co-operating across networks vs. ‘go-it-alone’ approach

Topic 3- Major Court battles regarding violation of patents between corporate companies

Text Books:

1. Joe Tidd, John Bessant. Managing Innovation: Integrating Technological, Market and Organizational Change
2. Case Study Materials: To be distributed for class discussion

RAJIV GANDHI PROUDYOGIKI VISHWAVIDYALAYA, BHOPAL

New Scheme Based On AICTE Flexible Curricula

IV Semester Bachelor of Technology (B.Tech.)-Computer Science and Business System (CSBS)

CB-405 Design Thinking

	Leadership Oriented Learning (LOL)	

Nature of Course	Behavioral
Pre requisites	Completion of all units from Semesters 1, 2, 3 and 4

Course Terminal Objectives:	
1	Recognize the importance of DT
2	Explain the phases in the DT process
3	List the steps required to complete each phase in DT process
4	Apply each phase in the DT process
5	Use doodling and storytelling in presenting ideas and prototypes
6	Create value proposition statements as part of their presentations
7	Recognize how DT can help in functional work
8	Recognize how Agile and DT complement each other to deliver customer satisfaction

Course Enabling Objectives:
Upon completion of the course, students shall have ability to

1	Recognize the importance of Design Thinking	[U]
2	Identify the steps in the DT process	[C]
3	Recognize the steps in the empathize phase of DT	[C]
4	Identify the steps required to conduct an immersion activity	[C]
5	Conduct an immersion activity and fill up the DT question template	[AP]
6	Recognize the steps to create personas in the define phase of DT	[C]
7	Create personas in the define phase of DT	[AP]
8	Recognize the steps to create problem statements in the define phase of DT	[AP]

9	Define the problem statements in the define phase of DT	[E]
10	Recognize the steps in the ideate phase of DT	[C]
11	Apply the steps in the ideate phase of DT	[AP]
12	Recognize how doodling can help to express ideas	[U]
13	Recognize the importance storytelling in presenting ideas and protypes	[U]
14	Recognize the importance of the prototype phase in DT	[C]
15	Create a prototype	[AP]
16	Recognize the importance of service value proposition	[C]
17	Create a value proposition statement	[AP]
18	Recognize the best practices of the testing phase in DT	[U]
19	Test a prototype created through a DT process	[AP]
20	Recognize how DT can help in functional work	[E]
21	Recognize how Agile and DT complement each other to deliver customer satisfaction	[C]

Course Contents:	
Total Hours:	45 hours
Textbooks:	
	There are no prescribed texts for Semester 5 – there will be handouts and reference links shared.
Reference Books:	
1	Hooked by NirEyal
2	The Art of Creative Thinking by Rod Judkins
3	Start Up nation by Dan Senor and Saul singer
4	Start with Why by Simon Sinek
Web References:	
1	What is Design Thinking? Interaction Design Foundation
2	What are some of the good examples of design thinking? - Quora
3	Design thinking 101: Principles, Tools & Examples to transform your creative process
Online Resources:	
1	Understanding Design thinking WF NEN
2	Design Thinking and Innovation at Apple Wei Li
3	Stanford Webinar- Design Thinking = Method, Not Magic
4	Stanford Design Thinking Virtual Crash Course
5	So Many Uses- activity to spark creativity and design

Assessment Methods & Levels (based on Bloom's Taxonomy)			
Formative assessment (Max. Marks:20)			
Course Outcome	**Bloom's Level**	**Assessment Component**	**Marks**
	Apply	Defining problem statement	5
	Apply	Ideating solutions	5
	Apply	Creating a prototype	10
Summative Assessment based on End Semester Project			
Bloom's Level			
Understand Apply Analyze	Understand, Analyze, Apply Conduct and apply DT in the project.		50

Lesson Plan

Unit No	Objective	Bloom's Level	Content	Type of Class	Duration
1	Recognize the importance of Design Thinking	2	**Why is Design Thinking important for business?** Stories and examples will be used to introduce Design Thinking to the participants. We will use relevant stories and the following videos. 1. YouTube video: The Design Thinking Process – Sprouts (3.57 mins) 2. Leverage TCS-provided DT content to show the evolution of DT and why is important in present business environment. Can be a video. (2 mins) Lecturer to encourage the students to maintain their Satori slam book and capture their learning points in it.	Introduction and discussion	60 mins
1	Recognize the importance of Design Thinking	2	**Why is Design Thinking important for you?** Experiential activity **Products that you loved and**	Activity	90 mins

Unit No	Objective	Bloom's Level	Content	Type of Class	Duration
			hated: In this activity, learners will have to share about a product they like of disliked based on their experience. What would they need in a bad product to make it good?		
1	Identify the steps in the DT process	2	**What is DT?** Introduce the 5-Step Stanford Model using YouTube videos: The video will give a brief idea about the five steps: • Empathize (search for rich stories and find some love) • Define (user need and insights – their POV) • Ideate (ideas, ideas, ideas) • Prototype (build to learn) • Test (show, don't tell) Start all over and iterate the flow as much as possible	Lecture and demo	60 mins
1	Recognize the steps in the empathize phase of DT	2	**What is empathy?** Touch the target activity (Recap from Sem 2 Unit 4) Discussions in class Reference: FHIL | Stages of Design Thinking | EMPATHY (2:29 mins)	Activity	60 mins
1	Identify the steps required to conduct an immersion activity	1 and 2	**How to empathize?** Moccasin Walk activity for 1 hour to allow learners experience stepping into the shoes of another person. *This is an individual activity.* Sharing observations with the group. Suggest that students try this even in their free time away from studies.	Activity and lecture	90 mins
1	Identify the steps required to conduct an immersion activity	1 and 2	**Intro to Immersion Activity** Introduction to immersion activity through flowcharts and handouts and examples (to be provided by	Lecture	45 mins

Unit No	Objective	Bloom's Level	Content	Type of Class	Duration
			TCS DT Team) (steps and the question template: 1. We met; 2. We were amazed to realize that; 3. We wonder if this means 4. It would change the world if)		
1	Conduct an immersion activity and fill up the DT question template	3	**Immersion activity** Participants will be divided into four groups. Each group will need to visit any one of the following places to conduct an immersion activity. They need to interview people and fill up the DT question template (explained in the last class) 1. College cafeteria 2. College library 3. College sports facility 4. Transport facility near college	Practical	180 mins
2	Recognize the steps to create personas in the define phase of DT Create personas in the define phase of DT	2 3	**Creating personas** Start with YouTube videos explaining the process of persona creation: 1. Personas – What is a persona and how do I create one? (2019) https://www.youtube.com/watch?v=GNvLpfXCge8 Each group will create at least one persona based on the immersion study they conducted in the empathize stage (refer to the four question templates). The group can use A4 pages, colours and other props to create and display their respective persona. **Reference:** https://www.interaction-design.org/literature/article/personas-why-and-how-you-should-use-	Lecture and practical	120 mins

Unit No	Objective	Bloom's Level	Content	Type of Class	Duration
			them Lecturer to guide participants on getting the personas right (based on guidelines provided by TCS DT Team).		
2	Recognize the steps to create problem statements in the define phase of DT	2	**Problem statements** Session will begin with YouTube videos on how to define problem statements in the Define phase. 1. FHIL | Stages of Design Thinking | REFRAME (1:55 mins) Lecturer will provide examples of problem statements in class (based on handouts provided by TCS DT Team)	Lecture and demo	60 mins
2	Define the problem statements in the define phase of DT	3	**Defining problem statements** Group activity, in which each group will define the key problem statements (max three) for their lead personas. Each group will present while the remaining groups will do a peer review. Finally, lecturer will moderate/validate the problem statements (based on handouts provided by TCS DT Team)	Formative assessment	90 mins
3	Recognize the steps in the ideate phase of DT	1 and 2	**How to Ideate?** The session will start with YouTube videos:	Lecture and demo	60 mins

Unit No	Objective	Bloom's Level	Content	Type of Class	Duration
			1. FHIL | Stages of Design Thinking | IDEATE (1:54 secs) 2. What Is Six Thinking Hats? (Litmos Heroes) (1:58 secs) Lecturer to briefly tell them about the guidelines of ideating (to be provided by TCS DT Team)		
3	Apply the steps in the ideate phase of DT	3	**Ideation games** Game 1: Six Thinking Hats Game 2: Million-dollar idea	Activity	90 mins
3	Apply the steps in the ideate phase of DT	3	**Ideate to find solutions** Participants will work in their assigned groups to ideate solutions for the problem statements they identified (as continuation of immersion activity) applying ideation methods discussed in the previous session. They will get scores based on how well they can apply the ideation methods. Lecturers will observe the groups separately and assign them scores based on specific rubric (provided by the TCS DT Team).	Formative assessment	90 mins
3	Recognize how doodling can help to express ideas	1	**Let's doodle!** Participants will first watch a video on doodling: Doodling – how it can help in presenting ideas during ideate and protype phases After that, participants will complete an activity on doodling.	Demo and activity	60 mins
3	Recognize the importance storytelling in presenting ideas and protypes	1	**What is Storytelling in DT?** Activity- Research to find out about people who have used DT in providing solutions. Present their findings in forms of stories. (Recap from Unit- Sem-) Suggested topics to be provided by the TCS DT team.	Activity	120 mins

Unit No	Objective	Bloom's Level	Content	Type of Class	Duration
4	Recognize the importance of the prototype phase in DT	2	**Why is a Prototype important in Design Thinking?** The session will start with an activity to drive home the importance of creating a prototype in the design thinking process. As part of debrief of the activity, lecturer will share relevant examples and prototyping guidelines (provided by the TCS DT Team). Finally, the participants will watch two YouTube videos: 1. FHIL | Stages of Design Thinking | PROTOTYPE 2. Prototyping Phase - Design Thinking | Coursera https://www.coursera.org/lecture/patient-safety-project-planning/prototyping-phase-jVuQn	Activity and demo	60 mins
4	Create a prototype	3	**Prototype your idea** This is a group activity in which the participants will work in groups (created at the beginning of the course, in which they did immersion, persona creation, defining problem statement and ideating) to create prototypes based on the solutions they had identified. Lecturer to share feedback based on guidelines provided by the TCs DT team.	Formative assessment	180 mins
4	Recognize the importance of service value proposition Create a value proposition statement	2 3	**Value Proposition Statement** You Tube: What is Value Proposition (by Venture Well) (3:51 mins)? Lecturer to discuss the guidelines for creating a value proposition	Lecture	120 mins 1635 mins

Unit No	Objective	Bloom's Level	Content	Type of Class	Duration
			statement (to be provided by the TCS DT Team) Each group now needs to create value proposition statement for the solution they have suggested.		
4	Recognize the best practices of the testing phase in DT	1	**Testing in Design Thinking** Participants will first watch a YouTube video: FHIL | Stages of Design Thinking | TESTING After that lecturers will explain them the importance of Testing the prototype through stories (provided by the TCS DT Team). They will also explain how the loop works in DT between the Empathize and Testing phases.	Lecture	60 mins
	Test a prototype created through a DT process	3	**Test the Prototype** Each group needs to test their prototype created earlier and: 1. Document user feedback 2. Write down their inference from the feedback 3. Suggest next steps (the loop that happens in DT)	Activity	120 mins
4	Recognize how DT can help in functional work	1	**Role of DT in your work** Lecturer conducts a group/open house discussion on: "How DT can help me to become a better coder?" Lecturer needs to capture the key learning points in these discussions.	Discussion	60 mins
4	Recognize how Agile and DT complement each other to deliver customer satisfaction	1	Suggested session on: **How Agile and DT complement each other to deliver customer satisfaction**	Lecture	45 mins
4			**Share your Satori** Participants will be asked to share their Satori moments from the DT sessions	Reflection activity	60 mins

Unit No	Objective	Bloom's Level	Content	Type of Class	Duration
					33 hours
			Project **Option 1:** Each group needs to present a Prototype of how they can apply DT in their functional work or coding. Examples will be provided to explain what exactly they need to do. Option 2: Each group will apply DT to create a prototype to improve any existing product or service. For both options, groups need to complete all phases of the Stanford DT model and include the outputs of each phase in their presentation. Lecturers will evaluate the project based on the rubric provided by the TCS DT Team.		12 hours
				Total	**45 hours**

RAJIV GANDHI PROUDYOGIKI VISHWAVIDYALAYA, BHOPAL

New Scheme Based On AICTE Flexible Curricula

IV Semester Bachelor of Technology (B.Tech.)-Computer Science and Business System (CSBS)

CB-406 Operations Research

Introduction to OR:
Origin of OR and its definition. Concept of optimizing performance measure, Types of OR problems,Deterministic vs. Stochastic optimization, Phases of OR problem approach – problem formulation,building mathematical model, deriving solutions, validating model, controlling and implementingsolution.

Linear Programming:
Linear programming – Examples from industrial cases, formulation & definitions, Matrix form. Implicit assumptions of LPP.
Some basic concepts and results of linear algebra – Vectors, Matrices, LinearIndependence/Dependence of vectors, Rank, Basis, System of linear eqns., Hyperplane, Convex set,Convex polyhedron, Extreme points, Basic feasible solutions.
Geometric method: 2-variable case, Special cases – infeasibility, unboundedness, redundancy &degeneracy, Sensitivity analysis.
Simplex Algorithm – slack, surplus & artificial variables, computational details, big-M method,identification and resolution of special cases through simplex iterations.
Duality – formulation, results, fundamental theorem of duality, dual-simplex and primal-dualalgorithms.

Transportation and Assignment problems:
TP - Examples, Definitions – decision variables, supply & demand constraints, formulation, Balanced & unbalanced situations, Solution methods – NWCR, minimum cost and VAM, test for optimality(MODI method), degeneracy and its resolution.
AP - Examples, Definitions – decision variables, constraints, formulation, Balanced &unbalanced situations, Solution method – Hungarian, test for optimality (MODI method), degeneracy & its resolution.

PERT – CPM:
Project definition, Project scheduling techniques – Gantt chart, PERT & CPM, Determination ofcritical paths, Estimation of Project time and its variance in PERT using statistical principles, Conceptof project crashing/time-cost trade-off.

Inventory Control:
Functions of inventory and its disadvantages, ABC analysis, Concept of inventory costs, Basics ofinventory policy (order, lead time, types), Fixed order-quantity models – EOQ, POQ &Quantitydiscount models. EOQ models for discrete units, sensitivity analysis and Robustness, Special cases ofEOQ models for safety stock with known/unknown stock out situations, models under prescribedpolicy, Probabilistic situations.

Queuing Theory:
Definitions – queue (waiting line), waiting costs, characteristics (arrival, queue, service discipline) ofqueuing system, queue types (channel vs. phase).
Kendall's notation, Little's law, steady state behaviour, Poisson's Process & queue, Models with examples - M/M/1 and its performance measures; M/M/m and its performance measures; brief description aboutsome special models.

Simulation Methodology:
Definition and steps of simulation, random number, random number generator, Discrete EventSystem Simulation – clock, event list, Application in Scheduling, Queuing systems and Inventory systems.

Laboratory

1. Formulation of linear programming problems.
2. Solution of linear programming problem using graphical method with:
    i. Multiple constraints
    ii. Unbounded solution
    iii. Infeasible solution
    iv. Alternative or multiple solution
3. Enumeration of all basic solutions for linear programming problem.
4. Solution of linear programming problem with simplex method.
5. Problem solving using Big M method.
6. Problem solving using two phase method.
7. Solution on primal problem as well as dual problem.
8. Solution based on dual simplex method.
9. Verification of weak duality, strong duality and complementary slackness property.
10. Solution of transportation problem.
11. Solution of assignment problem.
12. Solution of integer programming problem using Branch and Bound method.
13. Solution of integer programming problem using Gomory's cutting plane method.
14. Simulation: Random number generation.
15. Monte Carlo method.
16. Performance measures for M/M/1 queuing model.
17. ABC analysis.
18. Inventory model.

Text Books:

1. *Operations Research: An Introduction.*H.A. Taha.

Reference Books:

1. *Linear Programming*. K.G. Murthy.
2. *Linear Programming*. G. Hadley.
3. *Principles of OR with Application to Managerial Decisions*. H.M. Wagner.
4. *Introduction to Operations Research*. F.S. Hiller and G.J. Lieberman.
5. *Elements of Queuing Theory*. Thomas L. Saaty.
6. *Operations Research and Management Science, Hand Book*: Edited By A. Ravi Ravindran.
7. *Management Guide to PERT/CPM*. Wiest& Levy.

Modern Inventory Management. J.W. Prichard and R.H. Eagle.