B.Tech CSE COURSE SYLLABUS

DEPT OF Computer Science & Engineering

BIT MESRA

COURSE INFORMATION SHEET

Course code: BE101
Course title: Biological Science for Engineers
Pre-requisite(s): NIL
Co- requisite(s): NIL
Credits: 2 L:2 T:0 P:0
Class schedule per week: 02
Class: B. Tech
Semester / Level: III-IV /First
Branch: All
Name of Teacher:

Course Objectives

This course enables the students to:

1.	Recognize and understand the basic cell biology, biomolecules, related metabolic pathways and applicable bioenergetics.
2.	Relate common biological phenomenon at molecular level.
3.	Describe the chemical nature of enzymes and mechanism of action for their function in biochemical reactions.
4.	Correlate the molecular methods of biological signal generation and propagation in living system.
5.	Comprehend the steps involved in common application of biotechnology such as applicable for creation of transgenics, stem cells, plant metabolites production, PCR, ELISA.

Course Outcomes

After the completion of this course, students will be able to:

CO1	Demonstrate an understanding of fundamental biochemical principles, such as the structure/function of biomolecules involved in living system.
CO2	Interpret the bio mechanism involved in signal generation and transmission.
CO3	Correlate the basic methods involved in common biotechnological application.
CO4	Apply and effectively communicate scientific reasoning and data involved in common biotechnological applications.

BE101 Biological Science for Engineers

Credit:2

Module-1:
Basic Cell Biology: Origin of life, Cell theory, Cell Structure and function, Biomolecules, Cell cycle and cell division, Biological Organization.
[5L]

Module-2:
Bioenergetics and Metabolism: Gibbs free energy and thermodynamics, aerobic and anaerobic respiration, Glycolysis, Krebs cycle and electron transport chain, Beta oxidation, Photosynthesis.
[6L]

Module-3:
Enzymes and its Application: Classification of enzymes, Structure and mechanism of enzyme action and uses of enzymes, factors affecting enzyme activity, Immobilization of enzymes and their application.
[5L]

Module-4:
Biological Signal Generation and Propagation: Nerve cell structure and signal propagation. Mechanism of vision and hearing, cell signaling, Circadian rhythm.
[6L]

Module-5:
Engineering Biological Systems and its Applications:
Central dogma of molecular biology, Methods in genetic engineering and application, PCR, ELISA and its application, stem cell and tissue engineering. Artificial Intelligence in Biology, Plant factory.
[6L]

Books Recommended
Recommended Text Book

1. Purves et al, (1998) Life: *The Science of Biology,* 4th Ed.
2. R. Dulbecco, The Design of Life.
3. Lehninger A, Principals of Biochemistry , 5th Ed

Reference Book

1. Stryer, L. (2002). *Biochemistry*. New York: W.H. Freeman.
2. K. Wilson & K.H. Goulding, (2006) *A biologist's guide to Principles and Techniques of Practical Biochemistry*.

Gaps in the syllabus (to meet Industry/Profession requirements)

POs met through Gaps in the Syllabus

Topics beyond syllabus/Advanced topics/Design

POs met through Topics beyond syllabus/Advanced topics/Design

Course Delivery methods
Lecture by use of boards/LCD projectors/OHP projectors
Tutorials/Assignments
Seminars
Mini projects/Projects
Laboratory experiments/teaching aids
Industrial/guest lectures
Industrial visits/in-plant training
Self- learning such as use of NPTEL materials and internets
Simulation

Course Outcome (CO) Attainment Assessment tools & Evaluation procedure Direct Assessment

Assessment Tool	% Contribution during CO Assessment
Mid Sem Examination Marks	25
End Sem Examination Marks	50
Assignment / Quiz (s)	10+10
Teacher's Assesment	5

Assessment Components	CO1	CO2	CO3	CO4
Mid Sem Examination Marks	√	√	√	√
End Sem Examination Marks	√	√	√	√
Quiz I	√	√	√	
Quiz II	√	√	√	

Indirect Assessment –

1. Student Feedback on Faculty
2. Student Feedback on Course Outcome

Mapping between Objectives and Outcomes

Mapping of Course Outcomes onto Program Outcomes

Course Outcome #	a	b	c	d	e	f	g	h	i	j	K
1	3	3	3	3	1	1	1	2	1	1	1
2	3	3	3	3	1	1	1	2	1	1	1
3	1	3	3	3		1	1	1		1	1
4	2	2	2	2		2	2	2		1	1

If satisfying< 34%=1, 34-66% =2, > 66% = 3

Mapping Between COs and Course Delivery (CD) methods			
CD	**Course Delivery methods**	**Course Outcome**	**Course Delivery Method**
CD 1	Lecture by use of boards/LCD projectors/OHP projectors	CO1, 2, 3, 4	CD1, CD2, CD3, CD8
CD 2	Tutorials/Assignments	CO1, 2, 3, 4	CD1, CD2, CD3, CD8
CD 3	Seminars		
CD 4	Mini projects/Projects		
CD 5	Laboratory experiments/teaching aids		
CD 6	Industrial/guest lectures		
CD 7	Industrial visits/in-plant training		
CD 8	Self- learning such as use of NPTEL materials and internets		
CD 9	Simulation		

COURSE INFORMATION SHEET

Course code: IT201
Course title: **Basics of Intelligent Computing**
Pre-requisite(s):
Co- requisite(s):
Credits: L:3 T:0 P: 0
Class schedule per week: 3
Class: B. Tech
Semester / Level: II/2
Branch: All

Course Objectives

This course enables the students:

A.	To know the basic functions of different AI branches.
B.	To understand the functionalities of IoT .
C.	To know the application of fuzzy logic.
D.	To understand the basic functionalities of a cloud based system.
E.	To find the basic functions of soft computing.

Course Outcomes:

After the completion of this course, students will be able to:

1.	Identify the difference between different branches of AI.
2.	Analyze a fuzzy based system.
3.	Design Neural Networks to solve problems.
4.	Analyze a problem in terms of ANN point of view.
5.	Identify the components of a cloud-based system.

SYLLABUS

Module I
Introduction
Definition of Computing, Conventional Computing vs. Intelligent Computing, Necessity of Intelligent Computing, Current trends in Intelligent Computing

AI Concepts
Introduction to AI, AI problems and Solution approaches, Fundamentals of problem solving using Search and Heuristics, Overview of Knowledge-base creation, and Intelligent Agents, Classification of AI.

(8 L)

Module II
Introduction to Soft Computing
Hard Computing vs. Soft Computing, Paradigms of Soft Computing, Real Life applications of Soft Computing

Fuzzy Logic
Classical Sets Vs Fuzzy Sets, Membership Functions, Fuzzy operations, Fuzzy Relations, Fuzzy Composition (Max-Min, Max-Product), Defuzzification, Fuzzy Inference System

Genetic Algorithm
Principle of Optimization, Traditional vs Evolutionary optimization, Genetic Algorithm: Working Cycle of GA, Encoding, Crossover, Mutation.(8 L)

Module III
Introduction to Artificial Neural Networks:
Biological Neuron to Artificial Neuron, Mc-Culloh Pitts Perceptron Model, Layer of Neurons, Activation Function, Artificial Learning, Types of Learning, Introduction to Back Propagation Networks, Applications of Neural Network.
(8L)

Module IV
Introduction to Cloud computing
Conventional Computing, Historical developments, Defining a Cloud, Cloud Computing reference model, Overview of Virtualization: Introduction, Types of cloud, Cloud Platforms: Amazon Web Services, Microsoft Azure, Cloud Applications (8L)

ModuleV
Introduction to IOT
The IoT Paradigm, Concept of Things, IoT Hardware, IoT Protocols, IoT Architecture,

enabling technologies of IoT, IoT Designing and its levels. (8L)

Text books:

1. Rich Elaine, Knight Kevin, Nair S. B. Artificial Intelligence, 3rd Edition, Tata Mc. Graw Hill.
2. Padhy N. P., Simon S. P. Soft Computing: With MATLAB Programming, Oxford University Press, 2015.
3. Buyya Raj Kumar, Vecchiola Christian &Selvi S.Thamarai , Mastering Cloud Computing, McGraw Hill Publication, New Delhi, 2013.
4. Madisetti Vijay and BahgaArshdeep, Internet of Things (A Hands-on-Approach), 1st Edition, VPT, 2014.

Reference Books:

Raj Pethuru and Raman AnupamaC.,The Internet of Things: Enabling Technologies, Platforms, and Use Cases, CRC Press.

Konar Amit, Computational Intelligence: Principles, Techniques and Applications, Springer.

Shivanandam and Deepa, Principles of Soft Computing, 2nd Edition, John Wiley and Sons, 2011.

Gaps in the syllabus (to meet Industry/Profession requirements): N/A

POs met through Gaps in the Syllabus: P10 will be met though report-writing/presentation-based assignment

Topics beyond syllabus/Advanced topics/Design: Teaching through paper

POs met through Topics beyond syllabus/Advanced topics/Design: Teaching through paper

CD #	Course Delivery methods
CD1	Lecture by use of boards/LCD projectors/OHP projectors
CD2	Tutorials/Assignments
CD3	Seminars/ Quiz (s)
CD4	Mini projects/Projects
CD5	Laboratory experiments/teaching aids
CD6	Industrial/guest lectures
CD7	Industrial visits/in-plant training
CD8	Self- learning such as use of NPTEL materials and internets
CD9	Simulation

Course Outcome (CO) Attainment Assessment tools & Evaluation procedure

Direct assessment

Assessment tool	% contribution during co assessment
Mid semester examination	25
End semester examination	50
Quiz (s)	10+10
Assignment	5

Assessment Components	CO1	CO2	CO3	CO4
Mid SEM Examination Marks	3	3	2	
End SEM Examination Marks	3	3	3	3
Assignment / Quiz (s)	3	3	3	2

If satisfying < 34% = 1, 34-66% = 2, > 66% = 3

Indirect Assessment

1. Student Feedback on Faculty
2. Student Feedback on Course Outcome

Mapping of Course Outcomes onto Program Outcomes

Course Outcome	Program Outcomes											
	1	2	3	4	5	6	7	8	9	10	11	12
1	3	2	2	1	1	1	2	1	1	1	1	1
2	2	3	2	1	1	2	1	1	3	1	2	1
3	3	1	3	3	2	1	1	2	1	1	1	1
4	2	3	1	1	1	1	2	1	1	1	1	1
5	1	2	1	1	3	1	1	1	2	1	1	1

MAPPING BETWEEN COURSE OUTCOMES AND COURSE DELIVERY METHOD

Course Outcomes	Course Delivery Method
CO1	CD1,CD6
CO2	CD1, CD6,CD7

CO3	CD1, CD2, CD3,CD6,CD7
CO4	CD1, CD3,CD6,CD7
CO5	CD1,CD2,CD3,CD4,CD5,CD7

COURSE INFORMATION SHEET

Course code: MA205
Course title: Discrete Mathematics
Pre-requisite(s):
Co- requisite(s):
Credits: L:3 T:1 P:0 C: 4
Class schedule per week: 3 Lectures, 1 tutorial
Class: I. M.Sc. /B. Tech
Semester / Level: 2
Branch:
Name of Teacher:

Course Objectives: This course enables the students to

1.	exposed to a wide variety of mathematical concepts that are used in the Computer Science discipline, which may include concepts drawn from the areas of Number Theory, Graph Theory and Combinatorics.
2.	come across a number of theorems and proofs. Theorems will be stated and proved formally using various techniques.
3.	gain the various graphs algorithms along with its analysis
4.	apply graph theory based tools in solving practical problems.

Course Outcomes: After the completion of this course, students will be able to

CO1.	to model and analyze computational processes using analytic and combinatorial methods
CO2.	solve the problems of graph theory using graph algorithms
CO3.	apply computer programs (e.g. SAGE) to study graphs.
CO4.	apply counting techniques to solve combinatorial problems and identify, formulate, and solve computational problems in various fields.
CO5.	apply graph theory in the areas of computer science, operation research, biology, chemistry, physics, sociology, and engineering

SYLLABUS

MA205 **Discrete Mathematics**

Module I

Mathematical logic and Mathematical Reasoning, Compound Statements, Propositional Equivalences, Predicates and Quantifiers, Methods of Proof, Mathematical Induction, Well-ordering principal, Recursive Definition and Algorithms.

[9L]

Module II

Recurrence Relations, Classification of Recurrence Relations and their solutions by Characteristic Root method, Generating function and their various aspects, Utility of Generating function in solving Recurrence Relations.

[9L]

Module III

Set, Operations on Set, Computer representation of Set, Relations, Properties/Classification of Relations, Closure operations on Relations, Matrix representation of Relations, Digraphs. Functions and their Representation, Classification of Functions, Warshall's algorithm, Discrete Numeric Functions, Growth of Functions, Big O, Big Q, Hash Function, Growth Functions.

[9L]

Module IV

Binary Operations, Groups, Product and Quotients of Groups, Semi group, Products and Quotients of Semi groups, Permutation Group, Composition of Permutation, Inverse Permutation, Cyclic Permutation, Transposition, Even and Odd Permutation, Coding of Binary Information and Error Correction, Decoding and Error Correction.

[9L]

Module V

Introduction to Graph, Graph Terminologies and their Representation, Connected & Disconnected graphs, Isomorphic Graph, Euler & Hamilton graphs. Introduction to Trees, Versatility of Trees, Tree traversal. Spanning Trees, Minimum Spanning Tree.

[9L]

Text Books:

1. **Mott, Joe L., Abraham Kandel, and Theodore P. Baker** Discrete Mathematics for ComputerScientists& Mathematicians, PHI, 2nd edition 2002.
2. **Swapan Kumar Chakraborty and BikashKanti Sarkar**: Discrete Mathematics, Oxford Univ.Publication, 2010.
3. **Kolman, Bernard, Robert C. Busby, and Sharon Ross.** Discrete mathematical structures,Prentice-Hall, Inc., 2003.

Reference Books:

1. **BikashKanti Sarkar and Swapan Kumar Chakraborty,** *Combinatorics and Graph Theory*, PHI,2016.
2. **Seymour Lipschuz and Mark Lipson**,*Discrete Mathematics*, Shaum's outlines, 2003.
3. **Liu, Chung Laung,** *Elements of Discretemathematis*, Mcgraw Hill, 2nd edition, 2001.
4. Bondy and Murty, Grapg Theory with Applications, American Elsevier,1979.
5. Robin J. Wilson, Introduction to Graph Theory, Pearson, 2010.

6. Course delivery methods	
Lecture by use of boards/lcd projectors/ohp projectors	√
Tutorials/assignments	√
Seminars	
Mini projects/projects	√
Laboratory experiments/teaching aids	
Industrial/guest lectures	
Industrial visits/in-plant training	
Self- learning such as use of nptel materials and internets	√
Simulation	

Course outcome (co) attainment assessment tools & evaluation procedure

Direct assessment

Assessment tool	% contribution during co assessment
Mid semester examination	25
End semester examination	50
Quiz (s)	10+10
Assignment	5

Assessment components	CO1	CO2	CO3	CO4	CO5
Mid semester examination	√	√	√		
End semester examination	√	√	√	√	√
Quiz (s)	√	√	√		
Assignment	√	√	√	√	

Indirect assessment –

1. Student feedback on course outcome

Mapping of course outcomes onto program outcomes

Course outcome	Program outcomes											
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	2	2	1	1	1	1	3	3	2	2
CO2	3	2	2	2	1	1	2	1	3	3	2	2
CO3	3	3	2	2	1	1	1	1	3	3	2	2
CO4	2	2	3	1	1	1	1	1	3	3	2	2

CO5	2	2	3	3	1	2	1	1	3	3	2	2

If satisfying< 34%=1, 34-66% =2, > 66% = 3

MAPPING BETWEEN COURSE OUTCOMES AND COURSE DELIVERY METHOD

Course Outcomes	Course Delivery Method
CO1	CD1,CD6
CO2	CD1, CD6,CD7
CO3	CD1, CD2, CD3,CD6,CD7
CO4	CD1, CD3,CD6,CD7
CO5	CD1,CD2,CD3,CD4,CD5,CD7

COURSE INFORMATION SHEET

Course code: EC203
Course title: Digital System Design
Pre-requisite(s): EC101 Basics of Electronics & Communication Engineering
Co- requisite(s):

Credits: L: 3 T:0 P:0 C:3
Class schedule per week: 3x1
Class: B. Tech
Semester / Level: III/02
Branch: ECE
Name of Teacher:

Course Objectives

This course enables the students to:

A.	Understand the basics of the digital electronics.
B.	Apply the knowledge of digital electronics to construct various digital circuits.
C.	Analyse the characteristics and explain the outputs of digital circuits.
D.	Evaluate and asses the application of the digital circuits.
E.	Design digital machine for simple computing and control.

Course Outcomes

After the completion of this course, students will be able to:

CO 1	Explain the concept of digital electronics.
CO 2	Apply the knowledge to produce digital electronics circuits.
CO 3	Analyse and categorize digital circuits.
CO 4	Justify the uses of different digital circuits.
CO 5	Schematize and demonstrate simple computing machines.

SYLLABUS

Module – 1:

Basics of Digital Electronics: Number representation, Binary number system, Number base conversion, Octal, Hexadecimal and BCD codes, binary Arithmetic, Logic gates, Introduction to VHDL and Verilog, VHDL Models, Logic Families: TTL, ECL, and CMOS Logic Circuits, Logic levels, voltages and currents, fan-in, fan-out, speed, power dissipation. Comparison of logic families.

Module – 2:

Simplification of Boolean functions: Boolean Algebra, Basic theorems and Properties, De Morgan's theorem, Canonical & Standard forms, Simplification of Boolean function using Karnaugh map, POS& SOP simplification, Prime implicant, NAND and NOR implementation,.

Module – 3:

Design of Combinational Circuits: Analysis and design procedure, Parity Generators and Checkers, Adders, Subtractors, Look ahead carry, Adder, 4-bit BCD adder/subtractor, Magnitude comparator, Decoders, Encoders, Multiplexers, De-multiplexers, , Design of 1 bit ALU for basic logic and arithmetic operations.

Module – 4:

Design of Sequential Circuits and Memories: Basic Latch, Flip-Flops (SR, D, JK, T and Master-Slave), Triggering of Flip Flops, Synchronous and asynchronous counters, Registers, Shift Registers, Memories and Programmable Logic design, Types of memories, Memory Expansion and its decoding, Programmable Logic Arrays (PLA), Programmable Array Logic (PAL)

Module – 5:

Design of simple computing machines: SAP-I concepts with stress on timing diagrams, Microinstructions, Fetch and Execution cycle variable machine cycle, Hardware control Matrix, Macroinstructions, Microprogramming , Bus concepts, Multiplexed Minimum system. Pipelining concepts.

Books recommended:

Textbooks:

1. "Digital Design", Morris Mano and Michael D. Ciletti ,5th edition PHI
2. "Digital System Design using VHDL", Charles H Roth, Thomson Learning

Reference books:

1. Digital computer Electronics AP Malvino, 3rd Edition Mc Graw Hill

Gaps in the syllabus (to meet Industry/Profession requirements): Hands-on-practical on microprocessor trainer Kit

POs met through Gaps in the Syllabus:N/A

Topics beyond syllabus/Advanced topics/Design:N/A

POs met through Topics beyond syllabus/Advanced topics/Design:N/A

Course Outcome (CO) Attainment Assessment Tools and Evaluation Procedure

Direct Assessment

Assessment Tools	% Contribution during CO Assessment
Continuous Internal Assessment	50
Semester End Examination	50

Continuous Internal Assessment	% Distribution
Mid semester examination	25
Two quizzes	20 (2×10)
Teacher's Assessment	5

Assessment Components	CO1	CO2	CO3	CO4	CO5
Continuous Internal Assessment					
Semester End Examination					

Indirect Assessment

1. Student Feedback on Faculty
2. Student Feedback on Course

Course Delivery Methods

CD1	Lecture by use of boards/LCD projectors/OHP projectors
CD2	Assignments
CD3	Laboratory experiments/Teaching aids/Seminars
CD4	Mini Projects
CD5	Industrial visits/in-plant training
CD6	Self- learning such as use of NPTEL materials and internets
CD7	Simulation

Mapping between Course Outcomes and Program Outcomes

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	2	3	3	1	1			3		
CO2	3	3	2	3	3	3	2			3		
CO3	3	3	2	3	3	3	2			3		
CO4	3	3	2	3	3	2	2			3		
CO5	3	3	2	3	3	2	2			3		

< 34% = 1, 34-66% = 2, > 66% = 3

Mapping between Course Outcomes and Course Delivery Method

Course Outcomes	Course Delivery Method
CO1	CD1, CD2, CD3, CD6, CD7
CO2	CD1, CD2, CD3, CD6, CD7
CO3	CD1, CD2, CD3, CD6, CD7

CO4	CD1, CD2, CD3, CD6, CD7
CO5	CD1, CD2, CD3, CD6, CD7

COURSE INFORMATION SHEET

Course code: CS201
Course title: **Data Structures**
Pre-requisite(s): Programming for Problem Solving
Co- requisite(s): Data Structure Lab
Credits: L: 3 T: 1 P: 0
Class schedule per week: 4
Class: B. Tech
Semester / Level: II/2
Branch: CSE/IT

Course Objectives

This course enables the students:

A.	To be familiar with basic techniques of algorithm analysis.
B.	To understand basic concepts about arrays, stacks, queues, linked lists, trees and graphs.
C.	To understand concepts of searching and sorting techniques.
D.	To implement various linear & non-linear data structures; and searching & sorting algorithms.
E.	To assess how the choice of data structures impacts the performance of a program.

Course Outcomes

After the completion of this course, students will be able to:

1.	Define various linear and non-linear data structures like stack, queue, linked list, tree and graph.
2.	Explain operations like insertion, deletion, traversal, searching, sorting etc. on various data structures.
3.	Design various data structures and their operations.
4.	Analyze the performance of data structure based operations including searching and sorting.
5.	Justify the choice of appropriate data structure as applied to specified problem definition.

SYLLABUS

Module I

Basic Concepts

Definition and basics of: Data Structure, ADT, Algorithms, Time and Space Complexity, Asymptotic Notations (O, θ, Ω), Time complexity computation of non-recursive algorithms (like Matrix addition, Selection sort – using step count), Array – basic operations, concept of multi-dimensional array, Polynomial operations using Array, Sparse Matrix.

(8L)

Module II

Stack and Queue

Stack ADT: basic operations, Queue ADT: basic operations, Circular Queue, Evaluation of Expressions, Another application or Mazing Problem.

(8L)

Module III

Linked List

Singly Linked List: concept, representation and operations, Circular Linked List, Polynomial and Sparse Matrix operations using LL, Doubly Linked List: basic concept.

(8L)

Module IV

Tree and Graph

Basic concepts and terminologies, Binary Search Tree and Heap, Disjoint Set, Graph: concept and terminologies, Concept of BFS, DFS, Spanning Tree, Connected Components.

(8L)

Module V

Searching and Sorting

Sequential Search and Binary Search, Insertion Sort, Heap Sort, Radix Sort,External Sorting: k-way merging approach.

(8L)

Text book:

1. SahniHorwitz,, Freed Anderson, Fundamentals of Data Structures in C, 2nd Edition (or latest) , University Press.(T1)

Reference books:

1. TharejaReema, Data Structures Using C, 2nd Edition, Oxford University Press.(R1)
2. Tanenbaum, Langsam, Augenstein, Data Structures using C, Pearson. (R2)

Gaps in the syllabus (to meet Industry/Profession requirements): N/A

POs met through Gaps in the Syllabus: P10 will be met though report-writing/presentation-based assignment

Topics beyond syllabus/Advanced topics/Design: Teaching through paper

POs met through Topics beyond syllabus/Advanced topics/Design: Teaching through paper

CD #	Course Delivery methods
CD1	Lecture by use of boards/LCD projectors/OHP projectors
CD2	Tutorials/Assignments
CD3	Seminars/ Quiz (s)
CD4	Mini projects/Projects
CD5	Laboratory experiments/teaching aids
CD6	Industrial/guest lectures
CD7	Industrial visits/in-plant training

CD8	Self- learning such as use of NPTEL materials and internets
CD9	Simulation

Course Outcome (CO) Attainment Assessment tools & Evaluation procedure

Direct Assessment

Assessment tool	% contribution during co assessment
Mid semester examination	25
End semester examination	50
Quiz (s)	10+10
Assignment	5

Assessment Components	CO1	CO2	CO3	CO4
Mid SEM Examination Marks	3	3	2	
End SEM Examination Marks	3	3	3	3
Assignment / Quiz (s)	3	3	3	2

If satisfying < 34% = 1, 34-66% = 2, > 66% = 3

Indirect Assessment

1. Student Feedback on Faculty
2. Student Feedback on Course Outcome

Mapping of Course Outcomes onto Program Outcomes

Course Outcome	Program Outcomes											
	1	2	3	4	5	6	7	8	9	10	11	12
1	3	2	2	1	1				1	1	2	1
2	2	3	2	1	1	1	1	1		1	1	
3	2	2		3	1				1	1	1	2
4	2	3	1	1	1	1						
5	2	2	1	1	3	1		1	1	1	1	1

If satisfying < 34% = 1, 34-66% = 2, > 66% = 3

MAPPING BETWEEN COURSE OUTCOMES AND COURSE DELIVERY METHOD

Course Outcomes	Course Delivery Method
CO1	CD1,CD6
CO2	CD1, CD6,CD7
CO3	CD1, CD2, CD3,CD6,CD7
CO4	CD1, CD3,CD6,CD7
CO5	CD1,CD2,CD3,CD4,CD5,CD7

COURSE INFORMATION SHEET

Course code: EE102
Course title: EE102 ELECTRICAL ENGINEERING LABORATORY
Pre-requisite(s): Physics, Fundamentals of Mathematics and Electrical Engineering.

Credits:	L	T	P
	0	0	3

Class schedule per week: 3

Course Overview: Concepts of measuring instruments, AC RLC series parallel circuit operation, resonance, KVL and KCL, circuit theorems, 3-phase star and delta connections, measurement of low and high resistance of D.C. machine, measurement of power by three voltmeter, three-ammeter methods, measurement of power of 3-phase induction motor by two-wattmeter method.

Course Objectives

This course enables the students :

A.	To describe students practical knowledge of active and passive elements and operation of measuring instruments
B.	To demonstrate electrical circuit fundamentals and their equivalent circuit models for both 1-φ and 3- φ circuits and use circuit theorems
C.	To establish voltage & current relationships with the help of phasors and correlate them to experimental results
D.	1. To conclude performance of 1 – Φ AC series circuits by resonance phenomena 2. To evaluate different power measurement for both 1-φ and 3- φ circuits

Course Outcomes

After the completion of this course, students will be able to:

1.	classify active and passive elements, explain working and use of electrical components, different types of measuring instruments;
2.	illustrate fundamentals of operation of DC circuits, 1-φ and 3- φ circuits and also correlate the principles of DC, AC 1-φ and 3- φ circuits to rotating machines like Induction motor and D.C machine.;
3.	measure voltage, current, power, for DC and AC circuits and also represent them in phasor notations;

4.	analyse response of a circuit and calculate unknown circuit parameters;
5.	recommend and justify power factor improvement method in order to save electrical energy.

SYLLABUS

LIST OF EXPERIMENTS :

1. **Name: Measurement of low & high resistance of DC shunt motor**
   Aim: (i) To measure low resistance of armature winding of DC shunt motor
   (ii) To measure high resistance of shunt field winding of DC shunt motor

2. **Name: AC series circuit**
   Aim: (i) To obtain current & voltage distribution in AC RLC series circuit and to draw phasor diagram
   (ii) To obtain power & power factor of single phase load using 3- Voltmeter method and to draw phasor diagram

3. **Name: AC parallel circuit**
   Aim: (i) To obtain current & voltage distribution in AC RLC parallel circuit and to draw phasor diagram
   (ii) To obtain power & power factor of single phase load using 3- Ammeter method and to draw phasor diagram

4. **Name: Resonance in AC RLC series circuit**
   Aim : (i) To obtain the condition of resonance in AC RLC series circuit
   (ii) To draw phasor diagram

5. **Name: 3 phase Star connection**
   Aim : (i) To establish the relation between line & phase quantity in 3 phase star connection
   (ii) To draw the phasor diagram

6. **Name: 3 phase Delta connection**
   Aim : (i) To establish the relation between line & phase quantity in 3 phase delta connection
   (ii) To draw phasor diagram

7. **Name: 3 phase power measurement**

Aim : (i) To measure the power input to a 3 phase induction motor using 2 wattmeter method

(ii) To draw phasor diagram

8. **Name: Self & mutual inductance**

Aim : To determine self & mutual inductance of coils

9. **Name: Verification of Superposition, Thevenin's and Reciprocity theorem**

Aim : (i) To verify Superposition theorem for a given circuit

(ii) To verify Thevenin's theorem for a given circuit

10. **Name: Verification of Norton's, Tellegen's and Maximum Power transfer theorem**

Aim : (i) To verify Norton's theorem for a given circuit

(ii) To verify Maximum Power transfer theorem for a given circuit

Gaps in the syllabus (to meet Industry/Profession requirements)

1. Application of principles of magnetic circuits to electrical machines like transformers, generators and motors
2. Visualize Phase sequence

POs met through Gaps in the Syllabus :a, b, c, g

Topics beyond syllabus/Advanced topics/Design

1. Assignment : Simulation of electrical circuits with dependent/independent sources by various techniques (Mesh current/Node Voltage/Thevenin's theorem/Norton's theorem/Maximum power transfer theorem etc.) using MATLAB/PSIM/C++ softwares
2. Active/reactive power calculation for 3 – Φ circuits

POs met through Topics beyond syllabus/Advanced topics/Design: e,f, i, j, k

Mapping of lab experiment with Course Outcomes

Experiment	Course Outcomes				
	1	2	3	4	5
1	3	3	3	2	
2	3	3	3	3	2
3	3	3	3	3	2
4	3	3	3	3	2
5	3	3	3	1	
6	3	3	3	1	
7	3	3	3	2	2
8	3	3	3	3	
9	3	3	3	2	
10	3	3	3	2	

3=High, 2=Medium, 1=Low

Course Delivery methods	
CD1	Lecture by use of boards/LCD projectors
CD2	Tutorials/Assignments
CD3	Mini projects/Projects
CD4	Laboratory experiments/teaching aids
CD5	Self- learning such as use of NPTEL materials and internets
CD6	Simulation

Course Outcome (CO) Attainment Assessment tools & Evaluation procedure

Direct Assessment

Assessment Tool	% Contribution during CO Assessment
(1) Progressive Evaluation (60)	
Day to Day performance & Lab files	30
Quiz (s)	10
Viva	20
(2) End Semester (40)	
Examination Experiment performance	30
Quiz	10
Grand Total	100

Assessment Compoents	CO1	CO2	CO3	CO4	CO5
Progressive Evaluation Marks					
End Semester Marks					

Indirect Assessment –

1. Student Feedback on Faculty
2. Student Feedback on Course Outcome

Mapping of Course Outcomes onto Course Objectives

Course Outcome #	Course Objectives			
	A	B	C	D
1	3	3	3	3
2	3	3	3	3
3	3	3	3	3
4	3	3	3	3
5	2	3	3	3

Mapping of Course Outcomes onto Program Outcomes

Course Outcome #	Program Outcomes											
	a	b	c	d	E	f	g	h	i	j	k	
1	3	3	3	3	3	1	3	3	3	3	3	
2	3	3	3	2	2	2	2	3	3	3	3	
3	3	3	3	2	2	2	2	2	3	3	2	
4	3	3	3	3	3	1	2	2	3	3	2	
5	3	3	3	3	3	2	3	3	3	3	3	

Mapping of Course Outcomes onto Program Educational Objectives

Course Outcome #	Program Educational Objectives			
	1	2	3	4
1	3	3	2	2
2	3	3	3	

3	3	3	3	2
4	3	3	3	
5	3	3	2	2

Mapping Between COs and Course Delivery (CD) methods

Course Outcome	**Course Delivery Method**
CO1	CD1,CD2,CD4, CD5
CO2	CD1,CD4,CD5
CO3	CD1,CD3,CD4,CD5,CD6
CO4	CD1,CD2,CD4, CD5
CO5	CD4, CD5

COURSE INFORMATION SHEET

Course code: EC204
Course title: Digital System design Lab
Pre-requisite(s): EC101 Basics of Electronics & Communication Engineering
Co- requisite(s):
Credits: L:0 T:0 P:3 C:1.5
Class schedule per week: 03
Class: B. Tech
Semester / Level: III/ 02
Branch: ECE
Name of Teacher:

Course Objectives

This course enables the students to:

1.	Understand the basics of logic gates, input, output, power supply and gates IC's.
2.	Apply the knowledge of digital electronics to construct combinational and sequential circuits.
3.	Analyse controlled digital circuits with different Boolean function.
4.	Evaluate combinational/sequential circuits and memories.
5.	Translate real world problems into digital logic formulations using VHDL.

Course Outcomes

After the completion of this course, students will be able to:

CO 1	Describe the knowledge of basic logic gates and their design using universal gates.
CO 2	Demonstrate the working of combinational and sequential circuits.
CO 3	Integrate and experiment with controlled digital circuits.
CO 4	Appraise combinational/sequential circuits and memories.
CO	Schematize, simulate and implement combinational and sequential

5	circuits to solve real world problems using VHDL systems.

SYLLABUS

List of experiments:

1. Design and implement a controlled CMOS Inverter.
2. To study and verify the truth table of NAND and EX-OR gate using IC 7400.
3. Design and implement SEVEN segment display unit.
4. Design and verify half adder and full Adder circuits using gates and IC 7483.
5. Design and implement a 3:8 Decoder.
6. Design and implement 8:3 priority encoder.
7. Design a 4 bit magnitude comparator using combinational circuits.
8. Design and implement 8:1 multiplexer and 1:4 demultiplexer.
9. Design ALU with functions of ADD, SUB, INVERT, OR, AND. XOR, INC, DEC and CMP.
10. Design and verify decade Counter.
11. Design a ROM (8X4) using decoder, gates and diodes.
12. Design of pre settable up/down counter.

Implement all the above experiments using VHDL platform and verify.

Books recommended:

Textbooks:

1. "Digital Design", Morris Mano and Michael D. Ciletti ,5th edition PHI
2. "Digital System Design using VHDL", Charles H Roth, Thomson Learning

Reference books:

2. Digital computer Electronics AP Malvino, 3rd Edition Mc Graw Hill

Gaps in the syllabus (to meet Industry/Profession requirements):

POs met through Gaps in the Syllabus:

Topics beyond syllabus/Advanced topics/Design:

POs met through Topics beyond syllabus/Advanced topics/Design:

Course Outcome (CO) Attainment Assessment Tools and Evaluation Procedure

Direct Assessment

Assessment Tools	% Contribution during CO Assessment
Continuous Internal Assessment	60
Semester End Examination	40

Continuous Internal Assessment	% Distribution
Day to day performance & Lab files	30
Quiz(zes)	10
Viva	20

Semester End Examination	% Distribution
Examination Experiment Performance	30
Quiz	10

Assessment Components	CO1	CO2	CO3	CO4	CO5
Continuous Internal Assessment					
Semester End Examination					

Indirect Assessment

1. Student Feedback on Faculty
2. Student Feedback on Course

Course Delivery Methods

CD1	Lecture by use of boards/LCD projectors/OHP projectors
CD2	Assignments
CD3	Laboratory experiments/Teaching aids/Seminars
CD4	Mini Projects
CD5	Industrial visits/in-plant training
CD6	Self- learning such as use of NPTEL materials and internets
CD7	Simulation

Mapping between Course Outcomes and Program Outcomes

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	
CO1	3	3	3	3	3	1	1	1		3	2	
CO2	3	3	3	3	3	1	1	1		3	1	
CO3	3	3	3	3	3	1	1	1		3	1	
CO4	3	3	3	3	3	1	1	1		3	1	
CO5	3	3	3	3	3	1	1	1		3	1	

< 34% = 1, 34-66% = 2, > 66% = 3

Mapping between Course Outcomes and Course Delivery Method

Course Outcomes	Course Delivery Method
CO1	CD1, CD3, CD6, CD7
CO2	CD1, CD3, CD6, CD7
CO3	CD1, CD3, CD6, CD7
CO4	CD1, CD3, CD6, CD7
CO5	CD1, CD3, CD6, CD7

COURSE INFORMATION SHEET

Course code: CS202
Course title: **Data Structures Lab**
Pre-requisite(s):
Co- requisite(s):
Credits: L: 0 T: 0 P: 3
Class schedule per week: 3
Class: B. Tech
Semester / Level: II
Branch: All

Course Objectives

This course enables the students:

A.	To assess how the choice of data structures and algorithm design methods impact the performance of programs.
B.	To choose the appropriate data structure and algorithm design method for a specified application.
C.	To solve problems using data structures such as linear lists, stacks, queues, hash tables, binary trees, heaps, binary search trees, and graphs and writing programs for these solutions.
D.	Analyse and compare the different algorithms

Course Outcomes

After the completion of this course, students will be able to:

1.	Be able to design and analyze the time and space efficiency of the data structure

2.	Analyze run-time execution of previous learned sorting methods, including selection, merge sort, heap sort and Quick sort
3.	Have practical knowledge on the applications of data structures
4.	Be capable to identity the appropriate data structure for given problem

SYLLABUS

1. Program to Find the Number of Elements in an Array
2. Develop and Implement a menu driven program in C for the following Array operations
    a. Creating Array of N Integer elements.
    b. Display of Array elements with suitable headings.
    c. Inserting an element (ELEM) at a given valid position (POS).
    d. Deleting an element at a given valid position (POS).
    e. Exit
3. Programs for Stack, Queues and Circular Queues using Arrays
4. Program to convert an Infix Expression into Postfix and Postfix Evaluation
5. Program to implement stack using arrays
6. Program to implement stack using linked list
7. Program to implement multiple stack in a single array
8. Program to convert infix notation to postfix notation using stacks
9. Program to implement queue using arrays
10. Program to implement queue using pointers
11. Program to reverse elements in a queue
12. Program to implement circular queue using arrays
13. Program to create add remove & display element from single linked list
14. Program to create add remove & display element from double linked list
15. Program to count number of nodes in linear linked list
16. Program to create add remove & display element from circular linked list
17. Programs to implement stack & queues using linked representation
18. Program to concatenate two linear linked lists
19. Program to accept a singly linked list of integers & sort the list in ascending order.
20. Program to reverse linked list
21. Program to represent polynomial using linked list
22. Program to add two polynomials using linked list
23. Program for the creation of binary tree, provide insertion & deletion in c

24. Program for pre-order, post-order & in-order traversals of a binary tree using non reccursive.
25. Program to count no, of leaves of binary tree
26. Program for implementation of B-tree (insertion & deletion)
27. Program for implementation of multi-way tree in c
28. Program for implementation of AVL tree
29. Program to implement bubble sort program using arrays
30. Program to implement merge sort using arrays
31. Program to implement selection sort program using arrays
32. Program to implement insertion sort program using arrays
33. Program to implement topological sort using arrays
34. Program to implement heap sort using arrays
35. Program to implement heap sort using pointers
36. Program to implement bubble sort program using pointers
37. Program to implement linear search using pointers
38. Program to implement binary search using pointers
39. Program to implement linear search using arrays
40. Program to implement binary search using arrays

Text books:

1. Baluja G S, "Data Structure through C", Ganpat Rai Publication, New Delhi, 2015.
2. Pai G A V, "Data Structures and Algorithms: Concepts, Techniques and Applications", 2nd Edn, Tata McGraw-Hill, 2008.
3. Horowitz E., Sahni S., Susan A., "Fundamentals of Data Structures in C", 2nd Edition, University Press, 2010.

Reference books:

1. Tremblay J. P., Sorenson P. G, "An Introduction to Data Structures with Applications", 2nd Edn, McGraw-Hill, Inc. New York, NY, USA.
2. Lipschutz Seymour, "Data Structures", 6th Edn, 9th Reprint 2008, Tata McGraw-Hill.
3. Drozdek Adam, "Data Structures and Algorithms in C++", Thomson Learning, New Delhi – 2007.
4. Feller J., Fitzgerald B., "Understanding Open Source Software Development", Pearson Education Ltd. New Delhi

Gaps in the syllabus (to meet Industry/Profession requirements): N/A

POs met through Gaps in the Syllabus: N/A

Topics beyond syllabus/Advanced topics/Design: through experiments involving design/modelling of device/circuits on advanced topics

POs met through Topics beyond syllabus/Advanced topics/Design: through experiments involving design/modelling of device/circuits on advanced topics

CD #	Course Delivery methods
CD1	Lecture by use of boards/LCD projectors/OHP projectors
CD2	Tutorials/Assignments
CD3	Seminars/ Quiz (s)
CD4	Mini projects/Projects
CD5	Laboratory experiments/teaching aids
CD6	Industrial/guest lectures
CD7	Industrial visits/in-plant training
CD8	Self- learning such as use of NPTEL materials and internets
CD9	Simulation

Course Outcome (CO) Attainment Assessment tools & Evaluation procedure

Direct Assessment

Assessment Tools	% Contribution during CO Assessment
Continuous Internal Assessment	60
Semester End Examination	40

Continuous Internal Assessment	% Distribution
Day to day performance & Lab files	30
Quiz(zes)	10
Viva	20

Semester End Examination	% Distribution
Examination Experiment Performance	30
Quiz	10

Assessment Components	CO1	CO2	CO3	CO4
Progressive Evaluation	3	3	3	3
End SEM Evaluation	3	3	3	3

If satisfying < 34% = 1, 34-66% = 2, > 66% = 3

Indirect Assessment

1. Student Feedback on Faculty
2. Student Feedback on Course Outcome

MAPPING BETWEEN COURSE OUTCOMES AND PROGRAM OUTCOMES

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	3	3	2	1	2	1	1	2	1	1
CO2	3	3	3	3	2	1	2	2	2	1	1	
CO3	3	3	3	3	2	1	2	1	1	1	1	2
CO4	3	3	3	3	1	1	1	1	1	1	1	
CO5	3	3	3	3	2	1	1	2	1	1	1	1

If satisfying and < 34% = 1, 34-66% = 2, > 66% = 3

MAPPING BETWEEN COURSE OUTCOMES AND COURSE DELIVERY METHOD

Course Outcomes	Course Delivery Method
CO1	CD1,CD6
CO2	CD1, CD6,CD7
CO3	CD1, CD2, CD3,
CO4	CD1, CD3,CD6,CD7
CO5	CD1,CD2,CD7

COURSE INFORMATION SHEET

Course code: MA 203
Course title: Numerical Methods
Pre-requisite(s): NIL
Co- requisite(s): —NIL
Credits: L: 2 T: 0 P: 0 C: 2
Class schedule per week: 2 Lectures
Class: B Tech
Semester / Level: 2
Branch: ALL
Name of Teacher:
Course Objectives: This course enables the students to

1.	derive appropriate numerical methods to solve algebraic and transcendental equations
2.	derive appropriate numerical methods to solve linear system of equations
3.	approximate a function using various interpolation techniques
4.	to find the numerical solution of initial value problems and boundary value problems

Course Outcomes: After the completion of this course, students will be able to

CO 1	solve algebraic and transcendental equation using an appropriate numerical method arising in various engineering problems
CO 2	solve linear system of equations using an appropriate numerical method arising in computer programming, chemical engineering problems etc.
CO 3.	Approximate a function using an appropriate numerical method in various research problems
CO 4	evaluate derivative at a value using an appropriate numerical method in various research problems
CO 5	solve differential equation numerically

Syllabus

MA 203 Numerical Methods 2-0-0-2

Module I: Errors and Nonlinear Equations

Error Analysis: Definition and sources of errors, propagation of errors, floating-point arithmetic

Solution of Nonlinear equations: Bisection method, Regula-Falsi method, Secant method, Newton-Raphson method and its variants, General Iterative method. [05L]

Module II: System of Linear Equations

Gauss-Elimination, Gauss-Jordan, LU-Decomposition, Gauss-Jacobi and Gauss- Siedel methods to solve linear system of equations and Power method to find least and largest eigenvalues.

[05L]

Module III: Interpolation
Lagrange's interpolation, Newton's divided differences interpolation formulas, inverse interpolation, interpolating polynomial using finite differences.
[05L]

Module IV: Differentiation and Integration
Differentiation using interpolation formulas, Integration using Newton-Cotes formulas: Trapezoidal rule, Simpson's rule [05L]

Module V: Solution of Ordinary Differential Equations
Euler's method, modified Euler's method, Runge - Kutta Methods of second and fourth order to solve initial value problems.
[05L]

Text Books:
1. Jain M.K, S.R.K. Iyengar and R.K. Jain, Numerical Methods for Scientific and Engineering Computation, New Age Publications, 2004.
2. S.S. Sastry, Introductory Methods of Numerical Analysis, PHI.
3. E. Kreyszig, Advanced Engineering Mathematics, 9th Edition, John Wiley & Sons, 2006.

Reference Books:
1. S.C. Chapra and R. P. Canale, Numerical Methods for Engineers, McGraw Hill, 1985.
2. C.F. Gerald and P.O. Wheatley, Applied Numerical Analysis, Pearson Education, Seventh Edition, 2003.
3. R. W. Hamming: Numerical Methods for Scientists and Engineers, Second Edition, Dover

Course delivery methods	
Lecture by use of boards/lcd projectors/ohp projectors	√
Tutorials/assignments	√
Seminars	
Mini projects/projects	√
Laboratory experiments/teaching aids	
Industrial/guest lectures	
Industrial visits/in-plant training	

Self- learning such as use of nptel materials and internets	√
Simulation	

Course outcome (co) attainment assessment tools & evaluation procedure

Direct assessment

Assessment tool	% contribution during co assessment
Mid semester examination	25
End semester examination	50
Quiz (s)	10+10
Assignment	5

Assessment components	CO1	CO2	CO3	CO4	CO5
Mid semester examination	√	√	√		
End semester examination	√	√	√	√	√
Quiz (s)	√	√	√		
Assignment	√	√	√	√	

Indirect assessment –

1. Student feedback on course outcome

Mapping of course outcomes onto program outcomes

Course outcome	Program outcomes											
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	2	2	1	1	1	1	3	3	2	2
CO2	3	2	2	2	1	1	2	1	3	3	2	2
CO3	3	3	2	2	1	1	1	1	3	3	2	2

CO4	2	2	3	1	1	1	1	1	3	3	2	2
CO5	2	2	3	3	1	2	1	1	3	3	2	2

If satisfying< 34%=1, 34-66% =2, > 66% = 3.

MAPPING BETWEEN COURSE OUTCOMES AND COURSE DELIVERY METHOD

Course Outcomes	Course Delivery Method
CO1	CD1,CD6
CO2	CD1, CD6,CD7
CO3	CD1, CD2, CD3,
CO4	CD1, CD3,CD6,CD7
CO5	CD1,CD2,CD7

COURSE INFORMATION SHEET

Course code: CE101
Course title: **Environmental Science**
Pre-requisite(s):
Co- requisite(s):
Credits: L: 2 T: 0 P: 0
Class schedule per week: 2
Class: B. Tech
Semester / Level: I
Branch: All

Course Objectives

This course enables the students:

1	To develop basic knowledge of ecological principles and their applications in environment.
2	To identify the structure and composition of the spheres of the earth, the only planet sustaining life.
3	To analyse, how the environment is getting contaminated and probable control mechanisms for them.
4	To generate awareness and become a sensitive citizen towards the changing environment.

Course Outcomes

After the completion of this course, students will be:

1	Able to explain the structure and function of ecosystems and their importance in the holistic environment.
2	Able to identify the sources, causes, impacts and control of air pollution.
3	Able to distinguish the various types of water pollution happening in the environment and understand about their effects and potential control

	mechanisms.
4	Able to judge the importance of soil, causes of contamination and need of solid waste management.
5	Able to predict the sources of radiation hazards and pros and cons of noise pollution.

SYLLABUS

Module 1. Ecosystem and Environment

Concepts of Ecology and Environmental science, ecosystem: structure, function and services, Biogeochemical cycles, energy and nutrient flow, ecosystem management, fate of environmental pollutants, environmental status and reports on climate change.

(8L)

Module 2: Air Pollution

Structure and composition of unpolluted atmosphere, classification of air pollution sources, types of air pollutants, effects of air pollution, monitoring of air pollution, control methods and equipment for air pollution control, vehicular emissions and control, indoor air pollution, air pollution episodes and case studies.

(8L)

Module 3: Water Pollution

Water Resource; Water Pollution: types and Sources of Pollutants; effects of water pollution; Water quality monitoring, various water quality indices, water and waste water treatment: primary, secondary and tertiary treatment, advanced treatments (nitrate and phosphate removal); Sludge treatment and disposal.

(8L)

Module 4: Soil Pollution and Solid Waste Management

Lithosphere – composition, soil properties, soil pollution, ecological & health effects, Municipal solid waste management – classification of solid wastes, MSW characteristics, collection, storage, transport and disposal methods, sanitary landfills, technologies for processing of MSW: incineration, composing, pyrolysis.

(8L)

Module 5: Noise pollution & Radioactive pollution

Noise pollution: introduction, sources: Point, line and area sources; outdoor and indoor noise propagation, Effects of noise on health, criteria noise standards and limit values, Noise measurement techniques and analysis, prevention of noise pollution; Radioactive pollution: introduction, sources, classification, health and safety aspects, Hazards associated with nuclear reactors and disposal of spent fuel rods-safe guards from exposure to radiations, international regulation, Management of radioactive wastes.

(8L)

Text books:

1. A, K. De. (3rd Ed). 2008. Environmental Chemistry. New Age Publications India Ltd.
2. R. Rajagopalan.2016.Environmental Studies: From Crisis to Future by, 3rd edition, Oxford University Press.
3. Eugene P. Odum. 1971. Fundamentals of Ecology (3rd ed.) -. WB Sunders Company, Philadelphia.
4. C. N. Sawyer, P. L. McCarty and G. F. Parkin. 2002. Chemistry for Environmental Engineering and Science. John Henry Press.
5. S.C. Santra. 2011. Environmental Science. New Central Book Agency.

Reference books:

1. D.W.Conell. Basic Concepts of Environmental Chemistry, CRC Press.
2. Peavy, H.S, Rowe, D.R, Tchobanoglous, G. Environmental Engineering, Mc-Graw - Hill International
3. G.M. Masters& Wendell Ela. 1991. Introduction to Environmental Engineering and Science, PHI Publishers.

Gaps in the syllabus (to meet Industry/Profession requirements)

POs met through Gaps in the Syllabus

Topics beyond syllabus/Advanced topics/Design

POs met through Topics beyond syllabus/Advanced topics/Design

Course Delivery methods
Lecture by use of boards/LCD projectors/OHP projectors ✓
Tutorials/Assignments ✓
Seminars ✓
Mini projects/Projects
Laboratory experiments/teaching aids
Industrial/guest lectures
Industrial visits/in-plant training
Self- learning such as use of NPTEL materials and internets
Simulation

Course Outcome (CO) Attainment Assessment tools & Evaluation procedure

Direct Assessment

Assessment Tool	% Contribution during CO Assessment
Mid Sem Examination Marks	25
End Sem Examination Marks	50
Quiz (s) (1 & 2)	10+10
Teacher's assessment	5

Assessment Components	CO1	CO2	CO3	CO4	CO5
Mid sem exam	✔	✔	✔		
End Sem Examination Marks	✔	✔	✔	✔	✔
Assignment	✔	✔	✔	✔	✔

Indirect Assessment –

1.Student Feedback on Faculty
2. Student Feedback on Course Outcome

Mapping between Objectives and Outcomes

Mapping of Course Outcomes onto Graduate Attributes

Course Outcome #											
	a	b	c	d	e	f	g	h	i	j	k
1	2	2	2	2	L	3	3	2	2	2	1

2	2	3	2	3	2	3	3	2	2	2	1
3	2	3	2	3	2	3	3	2	2	2	1
4	2	3	2	3	2	3	3	2	2	2	1
5	2	3	3	3	2	3	3	2	2	2	1

If satisfying < 34% = 1, 34-66% = 2, > 66% = 3

Mapping Between Cos and Course Delivery (CD) methods				
CD	**Course Delivery methods**		**Course Outcome**	**Course Delivery Method**
CD 1	Lecture by use of boards/LCD projectors/OHP projectors		CO1	CD1, CD2
CD 2	Tutorials/Assignments		CO2	CD1, CD2
CD 3	Seminars		CO3	CD1, CD2
CD 4	Mini projects/Projects		CO4	CD1, CD2
CD 5	Laboratory experiments/teaching aids		CO5	CD1, CD2
CD 6	Industrial/guest lectures			
CD 7	Industrial visits/in-plant training			
CD 8	Self- learning such as use of NPTEL materials and internets			
CD 9	Simulation			

COURSE INFORMATION SHEET

Course code: CS203
Course title: **Computer Organization Architecture**
Pre-requisite(s): Digital Logic
Co- requisite(s):
Credits: L: 3 T: 1 P: 0
Class schedule per week: 4
Class: B. Tech
Semester / Level: II/2
Branch: CSE/IT

Course Objectives

This course enables the students:

1.	To understand the basic architecture and organization of systems along with their performances.
2.	To Familiar with Digital Logic circuits, Data representation and Instruction Set Architecture.

3.	To build a complete data path for various instructions.
4.	To understand the pipeline concepts and Hazards.
5.	To familiar with Memory and I/O Organization.

Course Outcomes

After the completion of this course, students will be to:

1.	Explain the merits and pitfalls in computer performance measurements and analyze the impact of instruction set architecture on cost-performance of computer design
2.	Explain Digital Logic Circuits ,Data Representation, Register and Processor level Design and Instruction Set architecture
3.	Solve problems related to computer arithmetic and Determine which hardware blocks and control lines are used for specific instructions
4.	Design a pipeline for consistent execution of instructions with minimum hazards
5.	Explain memory organization, I/O organization and its impact on computer cost /performance.

SYLLABUS

Module I

Basic Structures of Computers

Introduction to Digital Logic, Basic Structure of Computers: Computer Types, Functional Units, Input Unit, Memory Unit, Arithmetic and Logic Unit, Output Unit, Control Unit, Basic Operational Concepts: Fixed and floating point Representation and Arithmetic Operations, Performance, Historical Perspective.

(8L)

Module II

Instruction Set Architecture

Memory Locations and Addresses: Byte Addressability, Big-Endian and Little-Endian Assignments, Word Alignment, Instructions and Instruction Sequencing, Addressing

Modes, Assembly Language, Subroutines, Additional Instructions, Dealing with 32-Bit Immediate Values.

(8L)

Module III

Basic Processing Unit & Pipelining

Basic Processing Unit: Some Fundamental Concepts, Instruction Execution, Hardware Components, Instruction Fetch and Execution Steps, Control Signals, Hardwired Control, CISC-Style Processors.

Pipelining: Basic Concept, Pipeline Organization, Pipelining Issues, Data Dependencies, Memory Delays, Branch Delays, Pipeline Performance Evaluation.

(8L)

Module IV

Memory Organization

Basic Concepts, Semiconductor RAM Memories, Read-only Memories, Direct Memory Access, Memory Hierarchy, Cache Memories, Performance Considerations, Virtual Memory, Memory Management Requirements, Secondary Storage.

(8L)

Module V

Input Output & Parallel Processing

Basic Input Output

Accessing I/O Devices, Interrupts

Input Output Organization

Bus Structure, Bus Operation, Arbitration, Interface, Interconnection Standards.

Parallel Processing

Hardware Multithreading, Vector (SIMD) Processing, Shared-Memory Multiprocessors, Cache Coherence, Message-Passing Multicomputers, Parallel Programming for Multiprocessors, Performance Modeling.

(8L)

Text Book:

Patterson David A., Hennessy John L., Computer Organization and Design: The Hardware / Software Interface, 5th Edition, Elsevier.(T1)

Reference Books:

Hamachar Carl et. al , Computer Organization and Embedded Systems, 6th Edition, McGraw Hill. (R1)

Mano M. Morris, Computer System Architecture, Revised 3rd Edition, Pearson.(R2)

Gaps in the syllabus (to meet Industry/Profession requirements): N/A

POs met through Gaps in the Syllabus: P10 will be met though report-writing/presentation-based assignment

Topics beyond syllabus/Advanced topics/Design: Teaching through paper

POs met through Topics beyond syllabus/Advanced topics/Design: Teaching through paper

CD #	Course Delivery methods
CD1	Lecture by use of boards/LCD projectors/OHP projectors
CD2	Tutorials/Assignments
CD3	Seminars/ Quiz (s)
CD4	Mini projects/Projects
CD5	Laboratory experiments/teaching aids
CD6	Industrial/guest lectures
CD7	Industrial visits/in-plant training
CD8	Self- learning such as use of NPTEL materials and internets
CD9	Simulation

Course Outcome (CO) Attainment Assessment tools & Evaluation procedure

Direct Assessment

Assessment tool	**% contribution during co assessment**
Mid semester examination	25
End semester examination	50
Quiz (s)	10+10
Assignment	5

Assessment Components	CO1	CO2	CO3	CO4
Mid SEM Examination Marks	3	3	2	
End SEM Examination Marks	3	3	3	3
Assignment / Quiz (s)	3	3	3	2

If satisfying < 34% = 1, 34-66% = 2, > 66% = 3

Indirect Assessment

1. Student Feedback on Faculty
2. Student Feedback on Course Outcome

Mapping of Course Outcomes onto Program Outcomes

Course Outco	**Program Outcomes**											
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**

me												
1	3	3	3	2	2	1	1	1	1	1		
2	2	2				1	3	3	1	3		1
3	2	3	3	3	2	1	1					
4	2	2	3	3	2	1	1	2	2	1	1	
5	3	3	3	2	2	1	1	1	1	1	1	

If satisfying < 34% = 1, 34-66% = 2, > 66% = 3

MAPPING BETWEEN COURSE OUTCOMES AND COURSE DELIVERY METHOD

Course Outcomes	Course Delivery Method
CO1	CD1,CD6
CO2	CD1, CD6,CD7
CO3	CD1, CD2, CD3,CD6,CD7
CO4	CD1, CD3,CD6,CD7
CO5	CD1,CD2,CD3,CD4,CD5,CD7

COURSE INFORMATION SHEET

Course code: CS204
Course title: **Object Oriented Programming and Design Patterns**
Pre-requisite(s): Data Structure
Co- requisite(s):
Credits: L: 3 T: 0 P: 0
Class schedule per week: 3
Class: B. Tech
Semester / Level: II/2
Branch: CSE/IT

Course Objectives

This course enables the students:

1.	The course shall allow students to understand the basic tenets of OOP.
2.	The course will exemplify the basic syntax and constructs of JAVA.
3.	The course will help students understand the application OOP principles in various use cases.
4.	The course will explain basic JAVA GUI components and their working.
5.	The course aims to expose students to newer JAVA constructs like NIO, Lambdas etc.

Course Outcomes

After the completion of this course, students will be:

1.	Identify the difference between procedural and OO programming.
2.	Construct programs using various OOP principles.
3.	Design UI using JAVA GUI components.
4.	Operate on files and strings in real life scenarios.
5.	Analyze thread performance and inter thread communication issues

SYLLABUS

Module I

Introduction to Classes, Objects and Java

Introduction to Object Technology, Java, Understanding the Java development environment, Programming in Java, Memory concepts, Doing basic Arithmetic, Comparing entities, Classes, Objects, Methods, Strings, Primitive vs reference types.

(8L)

Module II

Control Statements, Methods and Arrays

Basic selection statements, Iterative constructs, Relative and Logical operators, break, continue, Methods, static methods, parameter passing, argument promotion and casting, scopes, method overloading. Arrays and ArrayList in Java, Enhanced for statement, Passing arrays to methods, Multidimensional arrays, Using command line arguments.

(8L)

Module III

Object Oriented Concepts: Polymorphism & Inheritance

Controlling access to class members, the use of this keyword, getters and setters, Composition, enum, the use of static and final, Garbage collection. Superclass and subclass, protected members, constructors in subclass, the Object class, Introduction to polymorphism, Abstract classes and methods, Assignment between subclass and superclass variables, Creating and using interfaces.

(8L)

Module IV

Exception Handling & GUI Design

When to use exception handling, Java exception hierarchy, finally block, Stack unwinding, Chained exceptions, Declaring new exception types, Assertions, try with resources. Simple I/O with GUI, Basic GUI Components, GUI Event handling, Adapter classes, Layout managers, Using panels.

(8L)

Module V

Strings, characters & Files

Working with the String and StringBuilder class, Character class, Tokenizing strings,

Regular Expressions, Files and Streams, Using NIO classes, Sequential file handling, Object serialization, JFileChooser, Introduction to threading, Introduction to Generics and lambda expressions.

(8L)

Text book:

Deitel P., Deitel H., Java How to Program, 10th Edition, Pearson Publications, 2016.(T1)

Reference book:

Wu C. T., Object Oriented Programming in Java, 5th Edition, McGrawHill Publications, 2010.(R1)

Gaps in the syllabus (to meet Industry/Profession requirements): N/A

POs met through Gaps in the Syllabus: P10 will be met though report-writing/presentation-based assignment

Topics beyond syllabus/Advanced topics/Design: Teaching through paper

POs met through Topics beyond syllabus/Advanced topics/Design: Teaching through paper

CD #	Course Delivery methods
CD1	Lecture by use of boards/LCD projectors/OHP projectors
CD2	Tutorials/Assignments
CD3	Seminars/ Quiz (s)
CD4	Mini projects/Projects
CD5	Laboratory experiments/teaching aids
CD6	Industrial/guest lectures
CD7	Industrial visits/in-plant training
CD8	Self- learning such as use of NPTEL materials and internets
CD9	Simulation

Course Outcome (CO) Attainment Assessment tools & Evaluation procedure

Direct Assessment

Assessment tool	% contribution during co assessment
Mid semester examination	25
End semester examination	50
Quiz (s)	10+10

Assignment	5

Assessment Components	CO1	CO2	CO3	CO4
Mid SEM Examination Marks	3	3	2	
End SEM Examination Marks	3	3	3	3
Assignment / Quiz (s)	3	3	3	2

If satisfying < 34% = 1, 34-66% = 2, > 66% = 3

Indirect Assessment

1. Student Feedback on Faculty
2. Student Feedback on Course Outcome

Mapping of Course Outcomes onto Program Outcomes

Course Outcome	Program Outcomes											
	1	2	3	4	5	6	7	8	9	10	11	12
1	3	1	1	2	3	1	1	1	1	2	2	1
2	2	3	1	1	1	1	1	1	1	1	2	
3	3	1	1	1	2	2	1	1	2	1	1	1
4	1	1	3	2	3	1	1	2	1	1	1	
5	3	1	1	1	1	1	1	1	1	2	1	1

MAPPING BETWEEN COURSE OUTCOMES AND COURSE DELIVERY METHOD

Course Outcomes	Course Delivery Method
CO1	CD1,CD6
CO2	CD1, CD6,CD7
CO3	CD1, CD2, CD3,CD6,CD7
CO4	CD1, CD3,CD6,CD7
CO5	CD1,CD2,CD3,CD4,CD5,CD7

COURSE INFORMATION SHEET

Course code: CS206
Course title: **Design and Analysis of Algorithm**
Pre-requisite(s): Data Structure
Co- requisite(s): Algorithms Lab
Credits: L: 3 T: 0 P: 0
Class schedule per week: 3
Class: B. Tech
Semester / Level: II/2
Branch: CSE/IT

Course Objectives

This course enables the students to:

1.	To analyze the performance of recursive and nor-recursive algorithms.
2.	To understand various algorithm design techniques.
3.	To use of different paradigms of problem solving.

4.	To find efficient ways to solve a given problem.
5.	To compare various algorithms of a given problem.

Course Outcomes

After the completion of this course, students will be able to:

1.	Define the concepts and mathematical foundation for analysis of algorithms.
2.	Explain different standard algorithm design techniques, namely, divide & conquer, greedy, dynamic programming, backtracking and branch & bound.
3.	Demonstrate standard algorithms for fundamental problems in Computer Science.
4.	Design algorithms for a given problem using standard algorithm design techniques.
5.	Analyze and compare the efficiency of various algorithms of a given problem.

SYLLABUS

Module I

Algorithms and Complexity

Introduction, Algorithm Complexity and various cases using Insertion Sort, Asymptotic Notations, Time complexity of Recursive Algorithm, Solving Recurrences using Iterative, Recursion Tree and Master Theorem. (8L)

Module II

Divide and Conquer

Discussion of basic approach using Binary Search, Merge Sort , Quick Sort , Selection in Expected linear time, Maximum Subarray , Matrix Multiplication , Introduction of Transform and Conquer and AVL Tree .

(8L)

Module III

Dynamic Programming

Introduction and Approach, Rod Cutting, LCS, Optimal BST, Transitive closure and All-pair Shortest Path, Travelling Salesperson Problem.

(8L)

Module IV
Greedy and other Design Approaches
Introduction to greedy using fractional knapsack, Huffman Code, Minimum Spanning Tree – Prim and Kruskal, Single Source Shortest Path Dijkstra's and Bellman-Ford, Introduction to Backtracking using N-Queens problem, Introduction to Branch and Bound using Assignment Problem or TSP.
(8L)

Module V
NP Completeness and Other Advanced Topics
Non-deterministic algorithms – searching and sorting, Class P and NP, Decision and Optimization problem, Reduction and NPC and NPH, NP Completeness proof for: SAT, Max-Clique, Vertex Cover, Introduction to Randomized Algorithms, Introduction to Approximation Algorithms.
(8L)

Text Book:
1. Cormen Thomas H. et al., Introduction to Algorithms. 3rd Edition, PHI Learning, latest edition.(T1)

Reference Books:
1 Horowitz E., Sahani, Fundamentals of Computer Algorithms, Galgotia Publication Pvt. Ltd. (R1)
2 Dave and Dave, Design and Analysis of Algorithms, 2nd Edition, Pearson. (R2)
3 Goodrich, Tamassia. Algorithm Design. Wiley. (R3)

Gaps in the syllabus (to meet Industry/Profession requirements): N/A

POs met through Gaps in the Syllabus: P10 will be met though report-writing/presentation-based assignment

Topics beyond syllabus/Advanced topics/Design: Teaching through paper

POs met through Topics beyond syllabus/Advanced topics/Design: Teaching through paper

CD #	Course Delivery methods
CD1	Lecture by use of boards/LCD projectors/OHP projectors
CD2	Tutorials/Assignments
CD3	Seminars/ Quiz (s)

CD4	Mini projects/Projects
CD5	Laboratory experiments/teaching aids
CD6	Industrial/guest lectures
CD7	Industrial visits/in-plant training
CD8	Self- learning such as use of NPTEL materials and internets
CD9	Simulation

Course Outcome (CO) Attainment Assessment tools & Evaluation procedure

Direct Assessment

Assessment tool	**% contribution during co assessment**
Mid semester examination	25
End semester examination	50
Quiz (s)	10+10
Assignment	5

Assessment Components	CO1	CO2	CO3	CO4
Mid SEM Examination Marks	3	3	2	
End SEM Examination Marks	3	3	3	3
Assignment / Quiz (s)	3	3	3	2

If satisfying < 34% = 1, 34-66% = 2, > 66% = 3

Indirect Assessment

1. Student Feedback on Faculty
2. Student Feedback on Course Outcome

Mapping of Course Outcomes onto Program Outcomes

Course Outcome	Program Outcomes											
	1	2	3	4	5	6	7	8	9	10	11	12
1	3	2	1	3	1	1	2	1	2	1	1	1
2	3	2	3	1	1	1	1	1	1	3	1	
3	1	3	2	2	2	1	1	2	1	1	1	
4	1	2	3	1	1	2	3	2	1	2	1	1
5	2	3	1	1	2	1	1	1	1		1	

MAPPING BETWEEN COURSE OUTCOMES AND COURSE DELIVERY METHOD

Course Outcomes	Course Delivery Method
CO1	CD1,CD6
CO2	CD1, CD6,CD7
CO3	CD1, CD2, CD3,CD6,CD7
CO4	CD1, CD3,CD6,CD7
CO5	CD1,CD2,CD3,CD4,CD5,CD7

Course code: CS303
Course title: **Operating System**
Pre-requisite(s): Data Structure, Computer System Architecture, Basic Course on Computer Programming
Co- requisite(s): None
Credits: L:3 T:0 P:0
Class schedule per week: 3
Class: BTech
Semester / Level: V
Branch: CSE/IT

Course Objectives

This course enables the students to:

1.	Present the main components of OS and their working
2.	Introduce the concepts of process and thread and their scheduling policies
3.	Handling synchronization of concurrent processes and deadlocks
4.	Analyze the different techniques for managing memory, I/O, disk and files
5.	Design the components of operating system

Course Outcomes

After the completion of the course student will be able to:

1.	Describe the main components of OS and their working
2.	Explain the concepts of process and thread and their scheduling policies
3.	Solve synchronization and deadlock issues
4.	Compare the different techniques for managing memory, I/O, disk and files
5.	Design components of operating system

Syllabus

Module I [8L]

Operating system Overview

Operating system Objective and Functions, Evolution of Operating System, Major Advances in OS Components, Characteristics of Modern Operating Systems

Process Description and Control

Process Concept, Process States, Process Description, Process Control, Threads, Types of Threads, Multicore and Multithreading

Module II **[8L]**

Scheduling

Type of scheduling, Uniprocessor Scheduling, Multiprocessor Scheduling

Module III **[8L]**

Concurrency

Mutual Exclusion and Synchronization

Principle of Concurrency, Mutual Exclusion, Hardware Support, Semaphores, Monitors, Message Passing, Readers/Writers Problem

Deadlock and Starvation

Principle of Deadlock, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Dining Philosopher Problem

Module IV **[8L]**

Memory Management

Memory Management Requirements, Memory Partitioning, Paging, Segmentation

Virtual Memory

Hardware and Control Structures, Operating System Policies for Virtual Memory

Module V **[8L]**

I/O Management and Disk Scheduling

I/O device, Organization of the I/O Function, Operating System Design Issues, I/O Buffering, Disk Scheduling, RAID, Disk Cache

File Management

Overview, File Organization and Access, File Directories, File Sharing, Record Blocking, File Allocation and Free Space Management

Text Book:

1. Stallings W., Operating systems - Internals and Design Principles, , 8th Edition, Pearson, 2014.

Reference Books:

1. Silberchatz Abraham , Galvin Peter B., Gagne Greg, Operating System Principles, 9th Edition, Wiley Student Edition, 2013.

2. Tanenbaum Andrew S. , Modern Operating Systems, 4th Edition, Pearson, 2014.

3. Dhamdhere D. M. , Operating Systems A concept - based Approach, 3rd Edition, McGrawHill Education, 2017.

4. Stuart B. L., Principles of Operating Systems, 1st Edition, 2008, Cengage learning, India Edition.

5. Godbole A. S., Operating Systems, 3rd Edition, McGrawHill Education, 2017.

Gaps in the syllabus (to meet Industry/Profession requirements): N/A

POs met through Gaps in the Syllabus: P10 will be met though report-writing/presentation-based assignment

Topics beyond syllabus/Advanced topics/Design: Teaching through paper

POs met through Topics beyond syllabus/Advanced topics/Design: Teaching through paper

CD #	Course Delivery methods
CD1	Lecture by use of boards/LCD projectors/OHP projectors
CD2	Tutorials/Assignments
CD3	Seminars/ Quiz (s)
CD4	Mini projects/Projects
CD5	Laboratory experiments/teaching aids
CD6	Industrial/guest lectures
CD7	Industrial visits/in-plant training
CD8	Self- learning such as use of NPTEL materials and internets
CD9	Simulation

Course Outcome (CO) Attainment Assessment tools & Evaluation procedure

Direct Assessment

Assessment tool	% contribution during co assessment
Mid semester examination	25
End semester examination	50
Quiz (s)	10+10

Assignment	5

Assessment Components	CO1	CO2	CO3	CO4
Mid SEM Examination Marks	3	3	2	
End SEM Examination Marks	3	3	3	3
Assignment / Quiz (s)	3	3	3	2

If satisfying < 34% = 1, 34-66% = 2, > 66% = 3

Indirect Assessment

1. Student Feedback on Faculty
2. Student Feedback on Course Outcome

Mapping of Course Outcomes onto Program Outcomes

Course Outcome	Program Outcomes										
	1	2	3	4	5	6	7	8	9	10	11
1	3	2	2	1	2	1	1	1	1	3	1
2	3	2	2	1	2	1	1	1	1	2	1
3	3	3	3	2	3	1	1	1	1	3	1
4	3	3	3	2	3	1	1	L	1	3	1
5	3	3	3	2	3	1	1	2	2	3	1

MAPPING BETWEEN COURSE OUTCOMES AND COURSE DELIVERY METHOD

Course Outcomes	Course Delivery Method
CO1	CD1,CD6
CO2	CD1, CD6,CD7
CO3	CD1, CD2, CD3,CD6,CD7
CO4	CD1, CD3,CD6,CD7
CO5	CD1,CD2,CD3,CD4,CD5,CD7

COURSE INFORMATION SHEET

Course code: CS304
Course title: Operating System Lab
Pre-requisite(s):
Co- requisite(s):
Credits: L: 0 T: 0 P: 3
Class schedule per week: 3
Class: B. Tech
Semester / Level: III
Branch: CSE/IT

Course Objectives
This course enables the students to:

1.	Gain practical experience with designing and implementing concepts of operating systems such as system calls.

2.	Implement and develop CPU scheduling.
3.	Implement and understand process management, memory management.
4.	To provide a foundation in use of file systems and deadlock handling using C language in Linux environment.

Course Outcomes

After the completion of this course, students will be able to:

CO1	Understand and implement basic services and functionalities of the operating system using system calls. .
CO2	Use modern operating system calls and synchronization libraries in software/ hardware interfaces.
CO3	Understand the benefits of thread over process and implement synchronized programs using multithreading concepts.
CO4	Analyze and simulate CPU Scheduling Algorithms like FCFS, Round Robin, SJF, and Priority.
CO5	Implement memory management schemes and page replacement schemes.

SYLLABUS

List of Programs as Assignments:

1. **Lab Assignment No: 1**

Objective: To Understand and Implement Directory Structure

Q1. WAP to create a File directory system.

2. **Lab Assignment No: 2**

Objective: To Understand and Implement Scheduling processes

Q1. WAP to schedule various processes

3. **Lab Assignment No: 3**

Objective: To Understand and Implement FCFS

Q1. WAP to implement FCFS CPU Scheduling

4. **Lab Assignment No: 4**

Objective: To Understand and Implement SJF

Q1. WAP to implement SJF CPU scheduling.

5. **Lab Assignment No: 5**

Objective: To Understand and Implement SRTF

Q1.WAP to implement SRTF CPU scheduling.

6. **Lab Assignment No: 6**

Objective: To Understand and Implement Scheduling algorithms

Q1. WAP to implement Round Robin Scheduling

7. **Lab Assignment No: 7**

Objective: To Understand and Implement Scheduling algorithms

Q1 WAP to implement SRTF scheduling .

8. **Lab Assignment No: 8**

Objective: To Understand and Implement context switching

Q1. WAP to implement Round Robin Scheduling with context switching.

9. **Lab Assignment No: 9**

Objective: To Understand and Implement context switching.

Q1.WAP to implement SRTF with context switching.

10. **Lab Assignment No: 10**

Objective: To Understand and Implement Page Replacement Techniques

Q1. WAP to implement FCFS page replacement algorithm.

Q2. WAP to implement Optimal page replacement algorithm.

Books recommended:

TEXT BOOKS

OPERATING SYSTEM CONCEPTS(2012): ABRAHAM SILBERSCHATZ Yale University PETER BAER GALVIN Pluribus Networks GREG GAGNE Westminster College. **(T1)**

Operating Systems (2003) by Deitel, Deitel, and Choffnes. (**T2**)

REFERENCE BOOKS

OPERATING SYSTEM CONCEPTS(2012): ABRAHAM SILBERSCHATZ Yale University PETER BAER GALVIN Pluribus Networks GREG GAGNE Westminster College. **(R1)**

Gaps in the syllabus (to meet Industry/Profession requirements): N/A

POs met through Gaps in the Syllabus: N/A

Topics beyond syllabus/Advanced topics/Design: through experiments involving design/modelling of device/circuits on advanced topics

POs met through Topics beyond syllabus/Advanced topics/Design: through experiments involving design/modelling of device/circuits on advanced topics

CD #	Course Delivery methods
CD1	Lecture by use of boards/LCD projectors/OHP projectors
CD2	Tutorials/Assignments
CD3	Seminars/ Quiz (s)
CD4	Mini projects/Projects
CD5	Laboratory experiments/teaching aids
CD6	Industrial/guest lectures
CD7	Industrial visits/in-plant training
CD8	Self- learning such as use of NPTEL materials and internets
CD9	Simulation

Course Outcome (CO) Attainment Assessment tools & Evaluation procedure

Direct Assessment

Assessment Tools	% Contribution during CO Assessment
Continuous Internal Assessment	60
Semester End Examination	40

Continuous Internal Assessment	% Distribution
Day to day performance & Lab files	30
Quiz(zes)	10
Viva	20

Semester End Examination	% Distribution
Examination Experiment Performance	30
Quiz	10

Assessment Components	CO1	CO2	CO3	CO4
Progressive Evaluation	3	3	3	3
End SEM Evaluation	3	3	3	3

If satisfying < 34% = 1, 34-66% = 2, > 66% = 3

Indirect Assessment

1. Student Feedback on Faculty
2. Student Feedback on Course Outcome

Mapping between Objectives and Outcomes

Mapping of Course Outcomes onto Program Outcomes

MAPPING BETWEEN COURSE OUTCOMES AND COURSE DELIVERY METHOD

Course Outcomes	Course Delivery Method
CO1	CD1,CD6
CO2	CD1, CD6,CD7
CO3	CD1, CD2, CD3,
CO4	CD1, CD3,CD6,CD7
CO5	CD1,CD2,CD7

Mapping between Objectives and Outcomes

Mapping of Course Outcomes onto Program Outcomes

Course Outcome	Program Outcomes					
	PO1	PO2	PO3	PO4	PO5	PO6
CO1	3	2	3	2	2	2

CO2	2	3	3	3	3	2
CO3	3	2	2	1	1	1
CO4	3	3	3	2	3	1
CO5	2	2	2	2	1	3

If satisfying and < 34% = 1, 34-66% = 2, > 66% = 3

MAPPING BETWEEN COURSE OUTCOMES AND COURSE DELIVERY METHOD

Course Outcomes	Course Delivery Method
CO1	CD1,CD4
CO2	CD1, CD2,CD5
CO3	CD1, CD2
CO4	CD1, CD3,CD5
CO5	CD1,CD2

COURSE INFORMATION SHEET

Course code: MA 204
Course title: Numerical Methods Lab
Pre-requisite(s): NIL
Co- requisite(s): —NIL
Credits: L: 0 T: 0 P: 2 C:1

Class schedule per week: 2 Sessionals
Class: BE
Semester / Level: III / UG
Branch: ALL
Name of Teacher:

Course Objectives

This course enables the students to understand

1.	derive appropriate numerical methods to solve algebraic, transcendental equations and linear system of equations
2.	approximate a function using various interpolation techniques, to find the numerical solution of initial value problems
3.	ncepts in probability theory, the properties of probability distributions
4.	estimation of mean, variance and proportion, the concepts of statistical hypothesis

Course Outcomes

After the completion of this course, students will be able to

1.	solve algebraic, transcendental equation and linear system of equations using an appropriate numerical method arising in various engineering problems
2.	evaluate derivative at a value using an appropriate numerical method in various research problems, solve differential equation numerically
3.	learn basic probability axioms, rules and the moments of discrete and continuous random variables as well as be familiar with common named discrete and continuous random variables.
4.	find the point and interval estimates, analyse data statistically and interpretation of the results

SYLLABUS

List of Assignments

1. Find a simple root of $f(x)=0$ using bisection method. Read the end points of the interval (a,b) in which the root lies, maximum number of iterations n and error tolerance eps.
2. Find a simple root of $f(x)=0$ using Regula-Falsi method. Read the end points of the interval (a,b) in which the root lies, maximum number of iterations n and error tolerance eps.
3. Find a simple root of $f(x)=0$ using Newton Raphson method. Read any initial approximation x_0, maximum number of iterations n and error tolerance eps.
4. Solution of a system of $n \times n$ linear equations using Gauss elimination method with partial pivoting. The program is for 10×10 system or higher order system.
5. Matrix inversion and solution of $n \times n$ system of equations using Gauss-Jordan method. If the system of equations is larger than 15×15 change the dimensions of the float statement.
6. Program to solve a system of equation using Gauss-Seidel iteration method. Order of the matrix is n, maximum number of iterations $niter$, error tolerance is eps and the initial approximation to the solution vector is x_0. If the system of equations is larger than 10×10 change the dimension in float.
7. Program to find the largest Eigen value in magnitude and the corresponding Eigen vector of a square matrix A of order n using power method.
8. Program for Lagrange interpolation.
9. Program for Newton divided difference interpolation.
10. Program for Newton's forward and backward interpolation.
11. Program for Gauss's central difference interpolation (both backward and forward).
12. Program to evaluate the integral of $f(x)$ between the limits a to b using Trapezoidal rule of integration based on n subintervals or $n+1$ nodal points. The values of a,b and n are to be read. The program is tested for $f(x)=1/(1+x)$.
13. Program to evaluate the integral of $f(x)$ between the limits a to b using Simpson's rule of integration based on $2n$ subintervals or $2n+1$ nodal points. The values of a,b and n are to be read and the integrand is written as a function subprogram. The program is tested for $f(x)=1/(1+x)$.
14. Program to solve an IVP, $dy/dx=f(x), y(x_0)=y_0$ using Euler method. The initial value x_0, y_0 the final value x_f and the step size h are to be read. The program is tested for $f(x,y)=-2xy^2$.
15. Program to solve an IVP, $dy/dx=f(x), y(x_0)=y_0$ using the classical Runge-Kutta fourth order method with step size h, $h/2$ and also computes the estimate of the truncation error. Input

parameters are: initial point, initial value, number of intervals and the step length h. Solutions

with h, $h/2$ and the estimate of the truncation error are available as output. The right hand

side The program is tested for $f(x,y) = -2xy^2$.

Text Books:

1. S.S.Sastry-Introductory Methods of Numerical Analysis-PHI, Private Ltd., New Delhi.
2. N.Pal& S. Sarkar- Statistics: Concepts and Applications, PHI, New Delhi-2005.

Reference Books:

1 R.V.Hogg et.al- Probability and Statistical Inpane, 7th Edn, Pearson Education, New Delhi-2006.

2. R.L.Burden&J.D.Faires- Numerical Analysis, Thomson Learning-Brooks/Cole, Indian Reprint, 2005.

Gaps in the syllabus (to meet Industry/Profession requirements): N/A

POs met through Gaps in the Syllabus: N/A

Topics beyond syllabus/Advanced topics/Design: through experiments involving design/modelling of device/circuits on advanced topics

POs met through Topics beyond syllabus/Advanced topics/Design: through experiments involving design/modelling of device/circuits on advanced topics

CD #	Course Delivery methods
CD1	Lecture by use of boards/LCD projectors/OHP projectors
CD2	Tutorials/Assignments
CD3	Seminars/ Quiz (s)
CD4	Mini projects/Projects
CD5	Laboratory experiments/teaching aids
CD6	Industrial/guest lectures
CD7	Industrial visits/in-plant training
CD8	Self- learning such as use of NPTEL materials and internets
CD9	Simulation

Course Outcome (CO) Attainment Assessment tools & Evaluation procedure

Direct Assessment

Assessment Tool	% Contribution during CO Assessment
Progressive Evaluation	(60)

Attendance Marks	12
Lab file Marks	12
Viva Marks	24
Day-to-day performance Marks	12
End SEM Evaluation	(40)
Lab quiz Marks	20
Lab performance Marks	20

Assessment Components	CO1	CO2	CO3	CO4
Progressive Evaluation	3	3	3	3
End SEM Evaluation	3	3	3	3

If satisfying < 34% = 1, 34-66% = 2, > 66% = 3

Indirect Assessment

1. Student Feedback on Faculty
2. Student Feedback on Course Outcome

Mapping between Objectives and Outcomes

Mapping of Course Outcomes onto Program Outcomes

Course Outcome		**Program Outcomes**										
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2	2	1	1	3						1	
CO2	2	1	1	1	3			3			1	
CO3	1	2	3	3	3						1	1
CO4		1	1	3	2					1		
CO5	1	1	2	2				2		3	1	1

If satisfying and < 34% = 1, 34-66% = 2, > 66% = 3

MAPPING BETWEEN COURSE OUTCOMES AND COURSE DELIVERY METHOD

Course Outcomes	Course Delivery Method
CO1	CD1,CD6
CO2	CD1, CD6,CD7
CO3	CD1, CD2, CD3,
CO4	CD1, CD3,CD6,CD7
CO5	CD1,CD2,CD7

COURSE INFORMATION SHEET

Course code: IT202
Course title: **Basic IT Workshop**
Pre-requisite(s):
Co- requisite(s):
Credits: L: 0 T: 0 P: 2
Class schedule per week: 2
Class: B. Tech
Semester / Level: IV/II
Branch: All

Course Objectives

This course enables the students:

1.	Understand and use the basic Matlab functions and understand its environment and variables
2.	Know about handling operations and advanced features like menus and toolbars
3.	Implement programs with the use of arrays, strings and graphical data representations
4.	Understand Python, Data Types, Operators, Arrays

5.	Implement Functions and loops, object oriented programming using Python

Course Outcomes

After the completion of this course, students will be able:

1.	Apply features of Matlab and algorithms to solve problems
2.	Develop application programs with the help of various tool boxes available in Matlab.
3.	Apply data analysis through graphical data representations
4.	Implement programs with the use of arrays, strings in Matlab
5.	Implement Functions and loops, using Python

Syllabus

Module I
Introduction to MATLAB and Basics Part I:

Introduction, Advantage, Disadvantage of MATLAB, MATLAB Environment, Variables and Array, Built-in Functions of MATLAB, Subarrays, Multidimensional Arrays, Data Files.

Module II

MATLAB Basic Part II:

Scalar and Array Operations, Hierarchy of Operations, Introduction to Plotting, Polar Plots, Subplots, MATLAB profiler. String Functions, Complex Data, Three-Dimensional Plot

Module III

MATLAB Advanced Features:

Sparse Arrays, Cell Arrays,Structure Arrays, I/O Functions, Object Handles, Position and Units, Graphical User Interface: Dialog Boxes, Menus, Toolbars.

Module IV

Introduction to Python Basics

Basics, I Python, Data Types, Operators, Arrays, Plotting

Module V

Python Programming Part 2:

Functions and loops, object oriented programming, Numerical Formalism

Sample list of Assignments:

Sample Assignments on Python

Data Types, Input- Outputs, Variables

1. Write a program in Python to swap two variables.

2. Write a program in Python to check the input character is an alphabet or not.

Loop

3. Write a program in python to shuffle a deck of card using the module random and draw 5 cards.

4. Write a program in python to find the factors of a number.

Array and Lists

5. Write a program in python to transpose a given matrix M = [[1, 2], [4, 5], [3, 6]].

6. Write a program in python to print the median of a set of numbers in a file.

Function

6. Write a function in Python to find the resolution of a JPEG image.

7. Write a program in python and use in-built functions to convert a decimal number to binary, octal and hexadecimal number.

8. Write a program in python to sort words in alphabetical order.

Plot

9. Use Matplotlib to draw histogram to represent average age of population given as Age [21, 54, 66, 44, 32,42, 54, 62, 93, 45, 32, 70]

10. Create a 3-D plot in Python for the function $\sqrt{y^2 - x^2}$ over the interval $-3 \leq x \leq 3$ and $-3 \leq x \leq 3$.

Sample Assignments on MATLAB

Assignment Statements:

1. Given two sides a= 3.2 and b=4.6 of a triangle and angle theta= 60^0 between these two sides. Find the length of the third side and the area of the triangle.

2. Write a MATLAB statement to calculate the sum of the series:

$$S= 1- x^2/2! + x^4/4! - x^6/6! + x^8/8! \quad \text{for } x= 1.5$$

Arrays

3. The array A is given below. Extend the 2-D array to 3-D array by including another 2-D array as second element in the third dimension.

A = 123;543;136;

4. Let a matrix A of size (3x4) is defined as, = 12356791011 4812 . Reshape the matrix A into matrix B of the size (6x2).

5. Let a column vector z be given as z = [2; 3; 4; 5].

(i) Form a diagonal matrix A, using the elements of z as he main diagonal elements of A.

(ii) Form the matrix B, using the elements of vector z as elements of upper diagonal of B.

(iii) Form the matrix C, using the elements of vector z as elements of first lower diagonal of C.

Polynomials

6. Integrate the polynomial $y = 4x^3 + 12x^2 + 16x + 1$. Take the constant of integration as 3.

7. Find the polynomial of degree 2 to fit the following data:

x	0	1	2	4
y	1	6	20	100

Input-Output statement and files

8. Write a program in MATLAB to illustrate the use of 'pause' command.

9. Write a program in MATLAB to illustrate the use of fwrite function for writing binary data of different formats to a file named 'check.txt'.

Plots

10. Plot the curve given by the equation y = sin(x) where x varies from 0 to 2π. Also label the x-axis and y-axis and provide a suitable title for the plot

11. Plot a bar graph for the data given as x = [1 2 3 4 5 6] and y = [10 15 25 30 27 19]

12. Given $x = t^2$ and y = 4t for -4 < t < 4. Using MATLAB obtain a 3-D plot showing the matrix in (x, y) space as a factors of time.

Control structures

13. Write a program in MATLAB to find the count of even values in the given n numbers.

Functions

14. Write a function in MATLAB to calculate the roots of the quadratic equation $ax^2 + bx + c = 0$, where a, b, c are constants.

Text Books:

1. MATLAB® Programming for Engineers:Stephen J. Chapman, Thomson Corporation, 4th Edition
2. Introduction to Python for Engineers and Scientists, Sandeep Nagar, Apress, 2018

Reference Books

1. Learn Python The Hard Way, Zed A. Shaw, Addison-Wesley, Third Edition

Gaps in the syllabus (to meet Industry/Profession requirements): N/A

POs met through Gaps in the Syllabus: N/A

Topics beyond syllabus/Advanced topics/Design: through experiments involving design/modelling of device/circuits on advanced topics

POs met through Topics beyond syllabus/Advanced topics/Design: through experiments involving design/modelling of device/circuits on advanced topics

CD #	Course Delivery methods
CD1	Lecture by use of boards/LCD projectors/OHP projectors
CD2	Tutorials/Assignments

CD3	Seminars/ Quiz (s)
CD4	Mini projects/Projects
CD5	Laboratory experiments/teaching aids
CD6	Industrial/guest lectures
CD7	Industrial visits/in-plant training
CD8	Self- learning such as use of NPTEL materials and internets
CD9	Simulation

Course Outcome (CO) Attainment Assessment tools & Evaluation procedure

Direct Assessment

Assessment Tools	% Contribution during CO Assessment
Continuous Internal Assessment	60
Semester End Examination	40

Continuous Internal Assessment	% Distribution
Day to day performance & Lab files	30
Quiz(zes)	10
Viva	20

Semester End Examination	% Distribution
Examination Experiment Performance	30
Quiz	10

Assessment Components	CO1	CO2	CO3	CO4
Progressive Evaluation	3	3	3	3
End SEM Evaluation	3	3	3	3

If satisfying < 34% = 1, 34-66% = 2, > 66% = 3

Indirect Assessment

1. Student Feedback on Faculty
2. Student Feedback on Course Outcome

Mapping between Objectives and Outcomes

Mapping of Course Outcomes onto Program Outcomes

MAPPING BETWEEN COURSE OUTCOMES AND COURSE DELIVERY METHOD

Course Outcomes	Course Delivery Method

CO1	CD1,CD6
CO2	CD1, CD6,CD7
CO3	CD1, CD2, CD3,
CO4	CD1, CD3,CD6,CD7
CO5	CD1,CD2,CD7

	Program Outcomes											
Course Outcome #	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**
CO1	3	3	3	1	3	2	1	3	1	2		
CO2	2	3	3	3	3	1	1	2	1	3		
CO3	1	3	2	1	3	1	1	1	1	1	1	
CO4	2	3	3	2	2	1	1	2	1	3		
CO5	3	3	1	2	3	1	1	2	1	1	1	

COURSE INFORMATION SHEET

Course code: CS205
Course title: **OOPDP Lab**
Pre-requisite(s):
Co- requisite(s):
Credits: L: 0 T: 0 P: 3
Class schedule per week: 3
Class: B. Tech

Semester / Level: IV/II
Branch: CSE/IT

Course Objectives

This course enables the students to:

1.	To introduce the student with fundamentals and features of Object Oriented programming.
2.	To be able to write a Java program to solve a well specified problem
3.	To be able to describe, recognise, apply and implement selected design patterns in Java
4.	To be familiar with common errors in Java and its associated libraries
5.	To understand a Java program written by someone else and be able to debug and test the same.

Course Outcomes

After the completion of this course, students will be able to:

CO1	Work in any object oriented environment and program using those features.
CO2	Student will have hands on experience with all basic concepts of Java programming
CO3	Analyse the design pattern of the given problem and further solve with less complexity.
CO4	Use his/her programming skills to resolve the issues coming while programming for bigger problems.
CO5	Work in industry environment with good enough knowledge about Java and OOPs.

SYLLABUS

List of Programs as Assignments:

1. Lab Assignment No: 1

Objective: To understand and Implement basic java program concepts using Scanner class.

Q1. Take input from user a character variable in a program and if the value is alphabet then print "Alphabet" if it's a number then print "Digit" and for other characters

print "Special Character"

Q2. Write a program to add all the values in a given number and check if the sum is prime number or not. Ex: 1234->10, not prime.

2. **Lab Assignment No: 2**

Objective: To Understand and Implement the concept of arrays in java

Q1. Write a program to find the largest 2 numbers and the smallest 2 numbers in the array initialized by the user.

Q2. Write a program to print the element of an array that has occurred the highest number of times Eg) Array -> 10,20,10,30,40,100,99 O/P:10

3. **Lab Assignment No: 3**

Objective: To Understand and Implement the concept of 2-D arrays in java.

Q1.Write a program to reverse the elements of a given 2*2 array. Four integer numbers needs to be passed as Command Line arguments

Eg: C:\>java Sample 1 2 3 4

O/P Expected :

The given array is : 1 2

3 4

The reverse of the array is : 4 3

2 1

Q2. Write a program to find greatest number in a 3*3 array. The program is supposed to receive 9 integer numbers as command line arguments.

4. **Lab Assignment No: 4**

Objective: To Understand and Implement the concept of classes and Constructors

Q1. Create a class Box that uses a parameterized constructor to initialize the dimensions of a box.(dimensions are width, height, depth of double type). The class should have a method that calculates and returns the volume of the box . Obtain an object and print the corresponding volume in main() function.

Q2. Write a program in Java with class Rectangle with the data fields width, length, area and color. The length, width and area are of double type and color is of string type. The methods are set_length() , set_width() , set_color(), and find_area(). Create two object ofRectangle and compare their area and color. If area and color same for the objects then display “Matching Rectangles” otherwise display “Non Matching Rectangle”.

5. **Lab Assignment No: 5**

Objective: To Understand and Implement the concept of Inheritance

Q1.Create a class named ‘Animal’ which includes methods like eat() and sleep(). Create a child class of Animal named ‘Bird’ and override the parent class methods. Add a new method named fly(). Create an instance of Animal class and invoke the eat and sleep methods using this object.Create an instance of Bird class and invoke

the eat, sleep and fly methods using this object.

Q2. A HighSchool application has two classes: the Person superclass and the Student subclass. Using inheritance, in this lab you will create two new classes, Teacher and CollegeStudent. A Teacher will be like Person but will have additional properties such as salary (the amount the teacher earns) and subject (e.g. “Computer Science”, “Chemistry”, “English”, “Other”). The CollegeStudent class will extend the Student class by adding a year (current level in college) and major (e.g. “Electrical Engineering”, “Communications”, “Undeclared”).

6. **Lab Assignment No: 6**

Objective: To Understand and Implement the concept of Overloading and Overriding

Q1.Create a class Account with two overloaded constructors. First constructor is used for initializing, name of account holder, account number and initial amount in account. Second constructor is used for initializing name of account holder, account number, address, type of account and current balance. Account class is having methods Deposit(), WithDraw(), and GetBalance().Make necessary assumption for data members and return types of the methods. Create objects of Account class and use them.

Q2. Create a base class Fruit which has name ,taste and size as its attributes. A method called eat() is created which describes the name of the fruit and its taste. Inherit the same in 2 other class Apple and Orange and override the eat() method to represent each fruit taste.

7. **Lab Assignment No: 7**

Objective: To Understand and Implement String class in Java

Q1. Reverse the string but not the words. Eg. I/P: Birla institute of technology

O/P: technology of institute birla.

Q2. Find out and print the maximum possible palindrome in a given string. Eg:

I/P: nononsene O/P: nonon

Q3.Given a string and a non-empty word string, return a string made of each char just before and just after every appearance of the word in the string. Ignore cases where there is no char before or after the word, and a char may be included twice if it is between two words.

If inputs are "abcXY123XYijk" and "XY", output should be "c13i".

If inputs are "XY123XY" and "XY", output should be "13".

8. **Lab Assignment No: 8**

Objective: To Understand and Implement the concept of Abstract classes and Interfaces

Q1. Create an abstract class Compartment to represent a rail coach. Provide an abstract function notice in this class. Derive FirstClass, Ladies, General, Luggage classes from the compartment class. Override the notice function in each of them to print notice suitable to the type of the compartment. Create a class

TestCompartment. Write main function to do the following: Declare an array of Compartment of size 10. Create a compartment of a type as decided by a randomly generated integer in the range 1 to 4. Check the polymorphic behavior of the notice method.

Q2.Write a program in java which implement interface Student which has two methods Display_Grade and Attendance for PG_Students and UG_Students (PG_Students and UG_Students are two different classes for Post Graduate and Under Graduate Students respectively).

9. **Lab Assignment No: 9**

Objective: To Understand and Implement Exception handling in java

Q1. Write a program in Java to display name and roll number of students. Initialize respective array variables for 10 students. Handle ArrayIndexOutOfBoundsException, so that any such problem does not cause illegal termination of program.

Q2. Write a program to accept name and age of a person from the command prompt(passed as arguments when you execute the class) and ensure that the age entered is >=18 and < 60. Display proper error messages. The program must exit gracefully after displaying the error message in case the arguments passed are not proper. (Hint : Create a user defined exception class for handling errors.)

10. **Lab Assignment No: 10**

Objective: To Understand and Implement File Handling and multithreading in java

Q1. Write a program to count the number of times a character appears in the Fileand also copy from one file to another. (Case insensitive... 'a' and 'A' are considered to be the same)

Q2. 1. Create class of SalesPersons as a thread that will display fives sales persons name. 2. Create a class as Days as other Thread that has array of seven days.

3. Call the instance of SalesPersons in Days and start both the threads 4. suspendSalesPersons on Sunday and resume on wednesday Note: use suspend, resume methods from thread

Q3. Create two threads, one thread to display all even numbers between 1 & 20, another to display odd numbers between 1 & 20. Note: Display all even numbers followed by odd numbers Hint: use join

11. **Lab Assignment No: 11**

Objective: To Understand and Implement Applets, AWT and Swings

Q1. Program to create a calculator with the help of AWT packages in Java.

Q2. Program to create a unit converter using Swings in Java.

Q3. APPLETS

a) Working with Frames and various controls.

b) Working with Dialogs and Menus.

c) Working with Panel and Layout.
d) Incorporating Graphics.
e) Working with colors and fonts

Books recommended:

TEXT BOOKS

1. Krishna P. R., Object Oriented Programming through JAVA, 1st Edition, Universities Press, 2008.
2. Patrick Naghton& H. Schildt – The Complete Reference Java 2, Tata McGraw Hill Publication, New Delhi.
3. Dietel,Dietel - Java How to program , 7th edition; Pearson Education , New Delhi.

REFERENCE BOOKS

1. C. Horstmann,G. Cornell - Core Java 2 Vol I & Vol II ; Pearson Education , New Delhi.
2. Balagurusamy -Programming in Java, 2nd Edition; Tata McGraw Hill Publication; New Delhi.

Course Evaluation:
Day to day progressive evaluation, Lab Quizzes, Surprise Tests, Online Lab performance and Viva Voce

Gaps in the syllabus (to meet Industry/Profession requirements):
Implementing of real world problems
POs met through Gaps in the Syllabus: PO5&6

Topics beyond syllabus/Advanced topics/Design:

POs met through Topics beyond syllabus/Advanced topics/Design: Teaching through research papers.

Gaps in the syllabus (to meet Industry/Profession requirements): N/A

POs met through Gaps in the Syllabus: N/A

Topics beyond syllabus/Advanced topics/Design: through experiments involving design/modelling of device/circuits on advanced topics

POs met through Topics beyond syllabus/Advanced topics/Design: through experiments involving design/modelling of device/circuits on advanced topics

CD #	Course Delivery methods
CD1	Lecture by use of boards/LCD projectors/OHP projectors

CD2	Tutorials/Assignments
CD3	Seminars/ Quiz (s)
CD4	Mini projects/Projects
CD5	Laboratory experiments/teaching aids
CD6	Industrial/guest lectures
CD7	Industrial visits/in-plant training
CD8	Self- learning such as use of NPTEL materials and internets
CD9	Simulation

Course Outcome (CO) Attainment Assessment tools & Evaluation procedure

Direct Assessment

Assessment Tools	% Contribution during CO Assessment
Continuous Internal Assessment	60
Semester End Examination	40

Continuous Internal Assessment	% Distribution
Day to day performance & Lab files	30
Quiz(zes)	10
Viva	20

Semester End Examination	% Distribution
Examination Experiment Performance	30
Quiz	10

Assessment Components	CO1	CO2	CO3	CO4
Progressive Evaluation	3	3	3	3
End SEM Evaluation	3	3	3	3

If satisfying < 34% = 1, 34-66% = 2, > 66% = 3

Indirect Assessment

1. Student Feedback on Faculty
2. Student Feedback on Course Outcome

Course Delivery Methods

CD1	Lecture by use of boards/LCD projectors/OHP projectors
CD2	Assignments/Seminars

CD3	Laboratory experiments/teaching aids
CD4	Industrial/guest lectures
CD5	Industrial visits/in-plant training
CD6	Self- learning such as use of NPTEL materials and internets
CD7	Simulation

Mapping between Objectives and Outcomes

Mapping of Course Outcomes onto Program Outcomes

Course Outcome	Program Outcomes											
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	2	2	1	1	3						1	
CO2	2	1	1	1	3			3				
CO3	1	2	3	3	3						1	
CO4		1	1	3	2					1		
CO5	1	1	2	2				2		3	1	

If satisfying and < 34% = 1, 34-66% = 2, > 66% = 3

MAPPING BETWEEN COURSE OUTCOMES AND COURSE DELIVERY METHOD

Course Outcomes	Course Delivery Method
CO1	CD1,CD6
CO2	CD1, CD6,CD7
CO3	CD1, CD2, CD3,
CO4	CD1, CD3,CD6,CD7
CO5	CD1,CD2,CD7

COURSE INFORMATION SHEET

Course code: CS207
Course title: **Design of Algorithm Lab**
Pre-requisite(s):
Co- requisite(s):
Credits: L: 0 T: 0 P: 3
Class schedule per week: 3
Class: B. Tech
Semester / Level: II
Branch: CSE/IT

Course Objectives

After the completion of this course, students will be able to:

1.	Able to implement various design strategies of algorithms
2.	Able to examine the efficiency of algorithm by changing the places of important steps.
3.	Able to compare approximate and exact solutions.
4.	Able to investigate effect randomness on correctness and efficiency of algorithms.
5.	Able to design approximate, random and parallel solution of different problems.

Course Outcomes

After the completion of this course, students will be able to:

CO 1	Able to know the different notions of asymptotic complexity and determine the asymptotic complexity of algorithms including the solving of recurrence

	relations.
CO 2	Able to determine the practical implications of asymptotic notations.
CO 3	Able to Implement, analyze, and compare algorithms.
CO 4	Able to Know the difference between the dynamic programming concept and a greedy approach.
CO 5	Able to know and use basic and advanced graph algorithms including DFS, BFS, and Bellman Ford.

SYLLABUS

List of Programs as Assignments:

1. **Lab Assignment No: 1**
   Programs on Polynomial vs logarithmic running times

2. **Lab Assignment No: 2**

Programs on Divide-and-conquer algorithms

3. **Lab Assignment No: 3**

Programs on Greedy and dynamic-programming algorithms

4. **Lab Assignment No: 4**

   Programs on Binary trees

5. **Lab Assignment No: 5**

Programs on Heaps and priority queues

6. **Lab Assignment No: 7**

Programs on Binary search trees

7. **Lab Assignment No: 8**

Programs on Hash tables

8. **Lab Assignment No: 9**

   Programs on Graph traversal

9. **Lab Assignment No: 10**

Programs on Shortest paths in graphs.

Books recommended:

Text Books:

1. Thomas H Cormen, Charles E Lieserson, Ronald L Rivest and Clifford Stein, Introduction to Algorithms, Second Edition, MIT Press/McGraw-Hill, 2001. **(T1)**
2. SanjoyDasgupta, Christos H. Papadimitriou and Umesh V. Vazirani, Algorithms, Tata McGraw-Hill, 2008. **(T2)**
3. Jon Kleinberg and ÉvaTardos, Algorithm Design, Pearson, 2005. **(T3)**

Course Evaluation:

Individual assignment, Theory (Quiz and End semester) examinations

Gaps in the syllabus (to meet Industry/Profession requirements): N/A

POs met through Gaps in the Syllabus: N/A

Topics beyond syllabus/Advanced topics/Design: through experiments involving design/modelling of device/circuits on advanced topics

POs met through Topics beyond syllabus/Advanced topics/Design: through experiments involving design/modelling of device/circuits on advanced topics

CD #	Course Delivery methods
CD1	Lecture by use of boards/LCD projectors/OHP projectors
CD2	Tutorials/Assignments
CD3	Seminars/ Quiz (s)
CD4	Mini projects/Projects
CD5	Laboratory experiments/teaching aids
CD6	Industrial/guest lectures
CD7	Industrial visits/in-plant training
CD8	Self- learning such as use of NPTEL materials and internets
CD9	Simulation

Course Outcome (CO) Attainment Assessment tools & Evaluation procedure

Direct Assessment

Assessment Tools	% Contribution during CO Assessment
Continuous Internal Assessment	60
Semester End Examination	40

Continuous Internal Assessment	% Distribution
Day to day performance & Lab files	30
Quiz(zes)	10
Viva	20

Semester End Examination	% Distribution
Examination Experiment Performance	30
Quiz	10

Assessment Components	CO1	CO2	CO3	CO4
Progressive Evaluation	3	3	3	3
End SEM Evaluation	3	3	3	3

If satisfying < 34% = 1, 34-66% = 2, > 66% = 3

Indirect Assessment

1. Student Feedback on Faculty
2. Student Feedback on Course Outcome

Mapping between Objectives and Outcomes

Mapping of Course Outcomes onto Program Outcomes

Course Outcome	Program Outcomes											
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	3	2	1	3	2					1	
CO2	3	3	2	2	1	1		3				
CO3	1	2	3	3	3						1	

CO4		2	2	1	1	1	1			1		
CO5	3	3	2	2				2		3	1	

If satisfying and < 34% = 1, 34-66% = 2, > 66% = 3

Mapping Between COs and Course Delivery (CD) methods			
CD	**Course Delivery methods**	**Course Outcome**	**Course Delivery Method**
CD1	Lecture by use of boards/LCD projectors/OHP projectors	CO1, CO2, CO3, CO4, CO5	CD1
CD2	Laboratory experiments/teaching aids		
CD3	Industrial/guest lectures		
CD4	Industrial visits/in-plant training		
CD5	Self- learning such as use of NPTEL materials and internets		