FACULTY OF ENGINEERING

Scheme of Instruction & Examination

(AICTE Model Curriculum for the Academic Year 2020-2021)

and

Syllabi

B.E. III and IV Semester

of

Four Year Degree Programme

in

Computer Engineering

(With effect from the academic year 2020–2021)
(As approved in the faculty meeting held on **-**-2020)

Issued by

Dean, Faculty of Engineering

Osmania University, Hyderabad – 500 007

2020

SCHEME OF INSTRUCTION & EXAMINATION
B.E. (Computer Engineering) III – SEMESTER

S. No.	Course Code	Course Title	Scheme of Instruction				Scheme of Examination			Credits
			L	T	P/D	Contact Hrs/Wk	CIE	SEE	Duration in Hrs	
Theory Courses										
1	HS204ME	Operations Research	3	-	-	3	30	70	3	3
2	BS206BZ	Biology for Engineers	3	-	-	3	30	70	3	3
3	BS205MT	Mathematical foundations for Data Science (Probability & Statistics)	3	-	-	3	30	70	3	3
4	ES214EC	Basic Electronics Engineering	3	-	-	3	30	70	3	3
5	ES216CM	Logic and Switching Theory	3	-	-	3	30	70	3	3
6	PC221CM	Data Structures	3	-	-	3	30	70	3	3
7	PC222CM	Discrete Structure & Mathematical Logic	3	-	-	3	30	70	3	3
Practical/ Laboratory Courses										
8	ES251EC	Basic Electronics Engineering Lab	-	-	2	2	25	50	3	1
9	PC252CM	Data Structures Lab	-	-	2	2	25	50	3	1
10	PC253CM	IT Workshop Lab	-	-	2	2	25	50	3	1
			21	**-**	**06**	**27**	**285**	**640**		**24**

HS: Humanities and Social Sciences BS: Basic Science ES: Engineering Science
MC: Mandatory Course PC: Professional Core
L: Lecture T: Tutorial P: Practical D: Drawing
CIE: Continuous Internal Evaluation SEE: Semester End Evaluation (Univ. Exam)
PY: Philosophy, BZ: Biology/ Life Sciences, CE: Civil Engineering, CS: Computer Science and Engineering
EC: Electronics and Communication Engineering, ME: Mechanical Engineering.

Note:

1. Each contact hour is a clock hour.

2. The duration of the practical class is two hours, however it can be extended wherever necessary, to enable the student to complete the experiment.

Course Code	Course Title						Core/Elective
HS204ME	**Operations Research**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

- Use variables for formulating complex mathematical models in management science, industrial engineering and transportation models.
- Use the basic methodology for the solution of linear programming problems.
- Understand the mathematical tools that are needed to solve optimization problems like Transportation models and Assignment models.
- Understand the replacement models with change in money value considering with time and without time.
- Model a system as a queuing model and compute important performance measures

Course Outcomes

After completing this course, the student will be able to:

1. Prepare the students to have the knowledge of Linear Programming Problem in Operations
2. Research at the end students would be able to understand the concept and develop the models for different applications.
3. Make students understand the concept Replacement models at the end students would able to explain various features and applications of replacement models in real time scenario.
4. Prepare the students to understand theory of Game in operations research at the end students would able to explain application of Game theory in decision making for a conflict
5. Prepare the students to have the knowledge of Sequencing model at the end student would able to develop optimum model for job scheduling.
6. Prepare students to understand Queuing theory concepts and various optimization techniques at the end students would able to develop models for waiting line cases.

UNIT-I

Introduction: Definition and Scope of Operations Research.

Linear Programming: Introduction, Formulation of linear programming problems, graphical method of solving LP problem, simplex method, maximization and minimization, Degeneracy in LPP, Unbounded and, Infeasible solutions.

UNIT-II

Duality: Definition, Relationship between primal and dual solutions, Economic Interpretation, Post optimal of sensitivity analysis, Dual Simplex Method.

UNIT-III

Transportation Models: Finding an initial feasible solution - North West corner method, least cost method, Vogel's Approximation method, Finding the optimal solution, optimal solution by stepping stone and MODI methods, Special cases in Transportation problems - Unbalanced Transportation problem.

Assignment Problems: Hungarian method of Assignment problem, Maximization in Assignment problem, unbalanced problem, problems with restrictions, travelling salesman problems.

UNIT-IV

Replacement Models: Introduction, replacement of items that deteriorate ignoring change in money value, replacement of items that deteriorate considering change in money value with time, replacement of items that fail suddenly - Individual replacement policy, Group replacement policy.

Game Theory: Introduction, 2 person zero sum games, Maximin - Minimax principle, Principle of Dominance, Solution for mixed strategy problems, Graphical method for 2 x n and m x 2 games.

UNIT-V

Sequencing Models: Introduction, General assumptions, processing n jobs through 2 machines, processing 'n' jobs through m machines, Processing 2 jobs through m machines

Queuing Theory: Introduction, single channel - Poisson arrivals - exponential service times with infinite population & finite population, Multi-channel - Poisson arrivals - Exponential service times with infinite population.

Introduction to Optimization Techniques: Single objective & Multi objective optimization Techniques like G.A, NSGA, P.Q.O & MPSO Techniques.

Suggested Readings:

1. Hamdy, A. Taha, Operations Research-An Introduction, Sixth Edition, Prentice Hall of India Pvt. Ltd., 1997.
2. S.D. Sharma, Operations Research, Kedarnath, Ramnath & Co., Meerut, 2009.
3. Hrvey M. Wagner, Principles of Operations Research, Second Edition, Prentice Hall of India Ltd., 1980.
4. V.K. Kapoor, Operations Research, S. Chand Publishers, New Delhi, 2004.
5. R. Paneer Selvam, Operations Research, Second Edition, PHI Learning Pvt. Ltd., New Delhi, 2008.
6. Data Reconciliation by Prof. Shanker Narasimha

Course Code	Course Title						Core/Elective
BS206BZ	**Biology for Engineers**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

Gain vivid knowledge in the fundamentals and uses of biology, human system and plant system.

Course Outcomes

After completing this course, the student will be able to:

1. Apply biological engineering principles, procedures needed to solve real-world problems.
2. Understand the fundamentals of living things, their classification, cell structure and biochemical constituents.
3. Apply the concept of plant, animal and microbial systems and growth in real life situations.
4. Comprehend genetics and the immune system.
5. Know the cause, symptoms, diagnosis and treatment of common diseases.
6. Apply basic knowledge of the applications of biological systems in relevant industries.

UNIT-I

Introduction to Life: Characteristics of living organisms, Basic classification, cell theory, structure of prokaryotic and eukaryotic cell, Introduction to Biomolecules: definition, general classification and important functions of carbohydrates, lipids, proteins, vitamins and enzymes.

UNIT-II

Biodiversity: Plant System: basic concepts of plant growth, nutrition, photosynthesis and nitrogen fixation. Animal System: Elementary study of digestive, respiratory, circulatory, excretory systems and their functions. Microbial System: History, types of microbes, economic importance and control of microbes.

UNIT-III

Genetics and Evolution: Theories of evolution and Evidences; cell division–mitosis and meiosis; evidence of laws of inheritance; variation and speciation; nucleic acids as a genetic material; central dogma; Mendel laws, gene and chromosomes.

UNIT-IV

Human Diseases: Definition, causes, symptoms, diagnosis, treatment and prevention of diabetes, cancer, hypertension, influenza, AIDS and Hepatitis. Immunity immunization, antigen – antibody immune response

UNIT-V

Biology and its Industrial Applications: Transgenic plants and animals, stem cell and tissue engineering, bioreactors, bio pharming, recombinant vaccines, cloning, drug discovery, biological neural networks, bioremediation, biofertilizer, biocontrol, biofilters, biosensors, biopolymers, bioenergy, biomaterials, biochips, basic biomedical instrumentation.

Suggested Readings:

1. A Text book of Biotechnology, R.C.Dubey, S. Chand Higher Academic Publications, 2013
2. Diseases of the Human Body, Carol D. Tamparo and Marcia A. Lewis, F.A. Davis Company, 2011.
3. Biomedical instrumentation, Technology and applications, R. Khandpur, McGraw Hill Professional, 2004
4. Biology for Engineers, Arthur T. Johnson, CRC Press, Taylor and Francis, 2011
5. Cell Biology and Genetics (Biology: The unity and diversity of life Volume I), Cecie Starr, Ralph Taggart, Christine Evers and Lisa Starr, Cengage Learning, 2008
6. Biotechnology Expanding horizon, B.D. Singh, Kalyani Publishers, 2012.

Course Code	Course Title						Core/Elective
BS205MT	**Mathematical Foundations for Data Science (Probability & Statistics)**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

- To introduce the solution methodologies for second order Partial Differential Equations with applications in engineering
- To provide an overview of probability and statistics to engineers

Course Outcomes

After completing this course, the student will be able to:

1. Solve field problems in engineering involving PDEs.
2. They can also formulate and solve problems involving random variables and apply statistical methods for analysing experimental data.

UNIT-I:Introduction of Probability, Conditional probability, Theorem of Total probability, Baye's Theorem and its applications, Random variables, Types of random variables, Probability mass function and Probability density function, Mathematical expectations.

UNIT-II: Discrete probability distributions: Binomial and Poisson distributions, Mean, variance, moment generating function and evaluation of statistical parameters for these distributions, Moments, Skewness and Kurtosis.

UNIT-III: Continuous probability distributions, Uniform, Exponential and Normal distributions, Mean, variance, moment generating function and evaluation of statistical parameters for these distributions

UNIT-IV: Curve fitting by the method of least squares: Fitting of straight lines, second degree parabolas and more general curves, Correlation, regression and Rank correlation. Test of significance: Large sample test for single proportion, difference of proportions, single mean, difference of means, and difference of standard deviations.

UNIT-V: Test for single mean, difference of means and correlation coefficients, test for ratio of variances, Chi-square test for goodness of fit and independence of attributes.

Suggested Readings:

1. R.K.Jain & Iyengar, "Advanced Engineering Mathematics", Narosa Publications.
2. B.S. Grewal, "Higher Engineering Mathematics", Khanna Publishers, 2000.
3. P.Sivaramakrishna Das & C.Vijaya Kumar, "Engineering Mathematics" , Pearson India Education Services Pvt. Ltd.
4. N.P. Bali & M. Goyal, "A Text Book of Engineering Mathematics", Laxmi Publications, 2010.
5. S.C.Gupta & V.K.Kapoor, "Fundamentals of Mathematical Statistics" , S.Chand Pub.
6. P. G. Hoel, S. C. Port & C. J. Stone, "Introduction to Probability Theory", Universal Book Stall, 2003.
7. W. Feller, "An Introduction to Probability Theory and its Applications", Vol. 1, Wiley, 1968.

Course Code	Course Title						Core/Elective
ES214EC	**Basic Electronics Engineering**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

The objectives of this course is to impart knowledge

- To analyze the behavior of semiconductor diodes in Forward and Reverse bias.
- To design of Half wave and Full wave rectifiers with L,C, LC & CLC Filters.
- To explore V-I characteristics of Bipolar Junction Transistor in CB, CE & CC configurations.
- To explain feedback concept and different oscillators.
- To analyze Digital logic basics and Photo Electric devices.

Course Outcomes

After completing this course, the student will be able to:

1. Able to learn about forward biased and reversed biased circuits.
2. Able to plot the V-I Characteristics of diode and transmission.
3. Able to design combinational logic circuits and PLDs.

UNIT-I

Semi-Conductor Theory: Energy Levels, Intrinsic and Extrinsic Semiconductors, Mobility, Diffusion and Drift current. Hall Effect, Characteristics of P-N Junction diode, Parameters and Applications.

Rectifiers: Half wave and Full wave Rectifiers (Bridge, center tapped) with and without filters, ripple regulation and efficiency. Zener diode regulator.

UNIT-II

Bipolar Junction Transistor: BJT, Current components, CE, CB, CC configurations, characteristics, Transistor as amplifier. Analysis of CE, CB, CC Amplifiers (qualitative treatment only) . JFET: Construction and working, parameters, CS, CG, CD Characteristics, CS amplifier.

UNIT-III

Feedback Concepts – Properties of Negative Feedback Amplifiers, Classification, Parameters. Oscillators – Barkhausen Criterion, LC Type and RC Type Oscillators and Crystal Oscillators. (Qualitative treatment only).

UNIT-IV

Operational Amplifiers – Introduction to OP Amp, characteristics and applications –Inverting and Non-inverting Amplifiers, Summer, Integrator, Differentiator, Instrumentation Amplifier. Digital Systems: Basic Logic Gates, half, Full Adder and Subtractors.

UNIT-V

Data Acquisition Systems: Study of transducer (LVDT, Strain gauge, Temperature, and Force). Photo Electric Devices and Industrial Devices: Photo diode, Photo Transistor, LED, LCD, SCR, UJT Construction and Characteristics only.

Display Systems: Constructional details of C.R.O and Applications.

Suggested Readings:

1. Jocob Millman, Christos C. Halkias and Satyabrata Jit,Electronics Devices and Circuits, 3rd Edition, McGraw Hill Education (India) Private Limited, 2010.
2. Rama Kanth A. Gaykward,Op-AMPS and Linear Integrated Circuit, 4th Edition PrenticeHall of India, 2000.
3. M. Morris Mano,Digital Design, 3rd Edition, Prentice Hall of India, 2002.
4.William D Cooper, and A.D. Helfrick, Electronic Measurements and Instrumentations Techniques, 2nd Edition, Prentice Hall of India, 2008.
5. S. Shalivahan, N. Suresh Kumar, A. Vallava Raj,Electronic Devices and Circuits, 2nd Edition., McGraw Hill Education (India) Private Limited, 2007.

Course Code	Course Title						Core/Elective
ES216EC	**Logic and Switching Theory**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives:

- To introduce concepts of Boolean logic, Postulates and Boolean Theorems.
- To understand the use of logic minimization methods and to solve the Boolean logic expressions
- To understand how to design the combinational and sequential circuits. • To introduce and realize the adder circuits
- To understand the state reduction methods for sequential circuits.

Course Outcomes:

Students will be

- Able to apply the concepts of Boolean logic, Postulates and Boolean Theorems to solve the Boolean expressions.
- Able to solve the Complex Boolean logic expressions using Minimization methods.
- Able to design the combinational, sequential circuits and Various adder circuits.
- Able to apply state reduction methods to solve sequential circuits.

UNIT-I

Boolean Algebra: Axiomatic definition of Boolean Algebra Operators, Postulates and Theorems, Boolean Functions, Canonical Forms and Standard Forms, Simplification of Boolean Functions Using Theorems and Karnaugh Map Method.

UNIT-II

Minimization of Switching Functions: Quine-McCluskey Tabular Method, Determination of Prime Implicants and Essential Prime Implicants. Combinational Logic Design: Single-Output and Multiple-Output

Combinational Circuit Design: AND-OR, OR-AND and NAND/NOR Realizations, Exclusive-OR and Equivalence functions.

UNIT–III

Design of Combinational Logic Circuits: Gate Level design of Small Scale Integration (SSI) circuits, Modular Combinational Logic Elements- Decoders, Encoders, Priority encoders, Multiplexers and De-multiplexers.

Design of Integer Arithmetic Circuits using Combinational Logic: Integer Adders – Binary Adders, Subtractors, Ripple Carry Adder and Carry Look Ahead Adder, and Carry Save Adders.

UNIT-IV

Design of Combinational Circuits using Programmable Logic Devices (PLDs): Programmable Read Only Memories (PROMs), Programmable Logic Arrays (PLAs), Programmable Array Logic (PAL) devices.

Introduction to Sequential Circuit Elements: Latch, Various types of Flip-Flops and their Excitation Tables.

UNIT -V

Models of Sequential Circuits: Moore Machine and Mealy Machine, Analysis of Sequential Circuits-State Table and State Transition Diagrams. Design of Sequential Circuits-Counters. Moore and Mealy State Graphs for Sequence Detection, Methods for Reduction of State Tables and State Assignments.

Suggested Reading:

1. M Morris Mano and Michael D Ciletti, Digital Design, Prentice Hall of India, Fourth Edition, 2008.
2. Zvi Kohavi, Switching and Finite Automata Theory, Tata McGraw Hill, 2nd Edition, 1979.
3. R. P Jain, Modern Digital Electronics,4th ed., McGraw Hill Education (India) Private Limited, 2003.
4. Ronald J.Tocci, Neal S. Widmer &Gregory L.Moss, "Digital Systems: Principles and Applications," PHI, 10/e, 2009.
5. Samir Palnitkar, "Verilog HDL A Guide to Digital Design and Synthesis," 2nd Edition, Pearson Education, 2006.

Course Code	Course Title						Core/Elective
PC221CM	**Data Structures**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

- To teach the importance of structuring the data for easy access and storage.
- To teach the implementation of various data structures.
- To acquire skills in using generic principles for data representation and manipulation with a view for efficiency, maintainability and code reuse.
- To introduce the basic concepts of advanced data structures.

Course Outcomes

After completing this course, the student will be able to:

1. Understand the importance of abstract data type and implementing the concepts of data structure using abstract data type.
2. Evaluate an algorithm by using algorithmic performance and measures.
3. Distinguish between linear and non-linear data structures and their representations in the memory using array and linked list.
4. Develop applications using Linear and Non-linear data structures.
5. Apply the suitable data structure for a real world problem and think critically for improvement in solutions.
6. Determine the suitability of the standard algorithms: Searching, Sorting and Traversals.

UNIT-I

Algorithms: Introduction, Algorithm Specifications, Recursive Algorithms, Performance Analysis of an algorithm- Time and Space Complexity, Asymptotic Notations, Complexity Analysis Examples.

Stacks and Queues: ADT Stack and its operations: Algorithms and their complexity analysis,

Applications of Stacks: Expression Conversion and evaluation –corresponding algorithms and complexity analysis.

Queue ADT and its operations: Linear Queue, Circular Queue, Algorithms and their analysis.

UNIT-II

Linked Lists: Singly linked lists: Representation in memory, Algorithms of several operations: Traversing, Searching, Insertion into, Deletion from linked list; Linked representation of Stack and Queue, Header nodes,

Doubly linked list: Operations on it and algorithmic analysis; Circular Linked Lists, Doubly Linked Lists, Applications (Polynomial Arithmetic).

Arrays and Matrices: Row And Column Major Representations, Sparse Matrices. Hashing: Hash Table Representation, Application- Text Compression.

UNIT- III

Trees: Definitions and Properties, Representation of Binary Trees, Operations, Binary Tree Traversal. Binary

Search Trees: Definitions, Operations on Binary Search Trees.

Balanced Search Trees: AVL Trees, Red Black Trees and B-Trees, Tree operations on each of the trees and their algorithms.

UNIT –IV

Graphs: Definitions and Properties, Representation, Graph Search Methods (Depth First Search and Breadth First Search)

Application of Graphs: Shortest Path Algorithm (Dijkstra), Minimum Spanning Tree (Prim's and Kruskal's Algorithms).

UNIT-V

Sorting and Searching: Objective and properties of different sorting algorithms: Selection Sort, Bubble Sort, Insertion Sort, Quick Sort, Merge Sort, Heap Sort, Linear and Binary Search algorithms.

Suggested Readings:

1. "Fundamentals of Data Structures in C++", Illustrated Edition by Ellis Horowitz, Sartaj Sahni, Dinesh Mehta, 2nd Edition, Universities Press.
2. Algorithms, Data Structures, and Problem Solving with C++", Illustrated Edition by Mark Allen Weiss, 3rd Edition, Pearson India.
3. "Fundamentals of Data Structures", Illustrated Edition by Ellis Horowitz, Sartaj Sahni, Computer Science Press.
4. "How to Solve it by Computer", 2nd Impression by R.G. Dromey, Pearson Education.
5. Michael T. Goodrich, Roberto Tamassia, David M. Mount, Data Structures and Algorithms in C++, John Wiley & Sons, 2010.

Course Code	Course Title						Core/Elective
PC222CM	**Discrete Structures & Mathematical logic**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

- To Learn mathematical concepts, terminology and notation as applied in computer science for solving logical problems.
- To Construct correct direct and indirect proofs.
- To Use division into cases in a proof.
- To Use counterexamples.
- Apply logical reasoning to solve a variety of problems
- To model relationships, analyse data, apply probability concepts and use functions to solve problems.
- To develop the mathematical skills needed for advanced quantitative courses.

Course Outcomes

After completing this course, the student will be able to:

1. For a given logic sentence express it in terms of predicates, quantifiers, and logical connectives
2. For a given a problem, derive the solution using deductive logic and prove the solution based on logical inference.
3. For a given a mathematical problem, classify its algebraic structure.
4. Evaluate Boolean functions and simplify expressions using the properties of Boolean algebra.
5. Develop the given problem as graph networks and solve with techniques of graph theory.

UNIT -I

Sets, Relation and Function: Operations and Laws of Sets, Cartesian Products, Binary Relation, Partial Ordering Relation, Equivalence Relation, Image of a Set, Sum and Product of Functions, Bijective functions, Inverse and Composite Function, Size of a Set, Finite and infinite Sets, Countable and uncountable Sets, Cantor's diagonal argument and The Power Set theorem, Schroeder-Bernstein theorem. **Principles of Mathematical Induction:** The Well-Ordering Principle, Recursive definition, The Division algorithm: Prime Numbers, The Greatest Common Divisor: Euclidean Algorithm, The Fundamental Theorem of Arithmetic.

UNIT-II

Basic counting techniques-inclusion and exclusion, pigeon-hole principle, permutation and combination.

UNIT-III

Propositional Logic: Syntax, Semantics, Validity and Satisfiability, Basic Connectives and Truth Tables, Logical Equivalence: The Laws of Logic, Logical Implication, Rules of Inference, The use of Quantifiers. **Proof Techniques:** Some Terminology, Proof Methods and Strategies, Forward Proof, Proof by Contradiction, Proof by Contraposition, Proof of Necessity and Sufficiency.

UNIT-IV

Algebraic Structures and Morphism: Algebraic Structures with one Binary Operation, Semi Groups, Monoids, Groups, Congruence Relation and Quotient Structures, Free and Cyclic Monoids and Groups, Permutation Groups, Substructures, Normal Subgroups, Algebraic Structures with two Binary Operation,

Rings, Integral Domain and Fields. Boolean Algebra and Boolean Ring, Identities of Boolean Algebra, Duality, Representation of Boolean Function, Disjunctive and Conjunctive Normal Form

UNIT–V

Graphs and Trees: Graphs and their properties, Degree, Connectivity, Path, Cycle, Sub Graph, Isomorphism, Eulerian and Hamiltonian Walks, Graph Colouring, Colouring maps and Planar Graphs, Colouring Vertices, Colouring Edges, List Colouring, Perfect Graph, definition properties and Example, rooted trees, trees and sorting, weighted trees and prefix codes, Bi-connected component and Articulation Points, Shortest distances.

Suggested books :

1. Kenneth H. Rosen, Discrete Mathematics and its Applications, Tata McGraw – Hill.

2. Susanna S. Epp, Discrete Mathematics with Applications,4th edition, Wadsworth Publishing Co. Inc.

3. C L Liu and D P Mohapatra, Elements of Discrete Mathematics A Computer Oriented Approach, 3rd Edition by, Tata McGraw – Hill.

Suggested reference books:

1. J.P. Tremblay and R. Manohar, Discrete Mathematical Structure and It's Application to Computer Science", TMG Edition, TataMcgraw-Hill

2. Norman L. Biggs, Discrete Mathematics, 2nd Edition, Oxford University Press. Schaum's Outlines Series, Seymour Lipschutz, Marc Lipson,

3. Discrete Mathematics, Tata McGraw - Hill

Course Code	Course Title						Core/Elective
ES251EC	**Basic Electronics Engineering Lab**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	-	-	-	**2**	**25**	**50**	**1**

Course Objectives

- To understand the characteristics of diodes and transistor configurations
- To understand the design concepts of biasing of BJT and FET
- To understand the design concepts of feedback amplifiers and oscillators
- To study the design concepts of OP Amp and data converters

Course Outcomes

After completing this course, the student will be able to:

1. Ability to design diode circuits & understand the application of Zener diode.
2. Ability to analyse characteristics of BJTs & FETs.
3. Ability to understand the different oscillator circuits.
4. Ability to understand operation of HWR & FWR circuits with & without filters.
5. Ability tom design Analog-to-Digital converters & Digital-to-Analog converters.

List of Experiments:

1. CRO-Applications, Measurements of R, L and C using LCR meter, Colour code method and soldering practice.
2. Characteristics of Semiconductors diode (Ge,Si and Zener)
3. Static Characteristics of BJT-Common Emitter
4. Static Characteristics of BJT-Common Base
5. Static Characteristics of FET
6. RC-Phase Shift Oscillator
7. Hartley and Colpitts Oscillators
8. Common Emitter Amplifier
9. Astable Multivibrator
10. Full-wave rectifier with and without filters using BJT
11. Operational Amplifier Applications
12. Strain Gauge Measurement
13. Analog-to-Digital and Digital to Analog Converters

Suggested Readings:

1. Maheshwari and Anand, *Laboratory Experiments and PSPICE Simulations in Analog Electronics,* 1st edition, Prentice Hall of India, 2006.
2. David Bell A., *Laboratory Manual for Electronic Devices and Circuits,* Prentice Hall of India, 2001.

Course Code	Course Title						Core/Elective
PC252CM	**Data Structures Lab**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	-	-	-	**2**	**25**	**50**	**1**

Course Objectives

- Design and construct simple programs by using the concepts of structures as abstract data type.
- To have a broad idea about how to use pointers in the implement of data structures.
- To enhance programming skills while improving their practical knowledge in data structures.
- To strengthen the practical ability to apply suitable data structure for real time applications.

Course Outcomes

After completing this course, the student will be able to:

1. Implement the abstract data type and reusability of a particular data structure.
2. Implement linear data structures such as stacks, queues using array and linked list.
3. Understand and implements non-linear data structures such as trees, graphs.
4. Implement various kinds of searching, sorting and traversal techniques and know when to choose which technique.
5. Understanding and implementing hashing techniques.
6. Decide a suitable data structure and algorithm to solve a real world problem.

1. Implementation of Stacks, Queues (using both arrays and linked lists).
2. Implementation of circular queue using arrays.
3. Implementation of double ended queue (de queue) using arrays.
4. Implement a program to evaluate a given postfix expression using stacks.
5. Implement a program to convert a given infix expression to postfix form using stacks.
6. Implement the following operations on singly linked list:
   i) Creation ii) Insertion iii) Deletion iv) Traversal
7. Implementation of Polynomial arithmetic using linked list.
8. Implement the following operations on doubly linked list:
   i) Creation ii) Insertion iii) Deletion iv) Traversal
9. Implement the following operations on circular linked list:
   i) Creation ii) Insertion iii) Deletion iv) Traversal
10. Implementation of recursive and iterative traversals on binary tree.
11. Implementation of operations on binary tree (delete entire tree, copy entire tree, mirror image, level order, search for a node etc.)
12. Implementation of the following operations on binary search tree (BST):
    (a) Minimum key (b) Maximum key (c) Search for a given key (d) Delete a node with given key
13. Implement the following operations on AVL search tree:
    i) Insertion ii) Deletion
14. Implement the following operations on B-Trees:
    i) Creation ii) Insertion iii) Deletion iv) Traversal
15. Implementation of graph traversals by applying: (a) BFS (b) DFS
16. Implement the following algorithms to find out a minimum spanning tree of a simple connected undirected graph: (a) Prim's algorithm (b) Kruskal's algorithm
17. Implement Dijkstra's algorithm for solving single source shortest path problem.
18. Implementation of recursive and non recursive functions to perform the following searching operations for a key value in a given list of integers: i) Linear search ii) Binary search
19. Implement the following sorting algorithms:
    a) Bubble sort b) Selection sort c) Insertion sort (d) Merge sort (e) Quick sort (f) Heap sort
20. Implementation of hashing with (a) Separate Chaining and (b) Open addressing methods.

Course Code	Course Title						Core/Elective
PC253CM	**IT Workshop Lab**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**-**	**-**	**-**	**2**	**25**	**50**	**1**

Course Objectives

- Introducing a new object oriented programming
- Enabling students to learn Big Data, Machine Learning etc.
- Preparing students to cope up with new Market tendencies
- To learn programs in MATLAB environment
- To handle Functions, Polynomials by using MATLAB commands
- Ability to solve any Mathematical functions
- To learn Mathematical Modelling in a new approach
- To plot Graphics (2-D) easily and effectively

Course Outcomes

After completing this course, the student will be able to:

1. Implement basic syntax in python.
2. Analyse and implement different kinds of OOP concept in real world problems.
3. Implement MATLAB operations and graphic functions.

SECTION 1 : MAT LAB / SCILAB PROGRAMS

1. Introduction to MATLAB/SCIlab Environment. MATLAB Menus, Toolbars, Computing with MATLAB, Script Files and the Editor/Debugger, MATLAB help System.

2. MATLAB controls: Relational Logical Variables. Conditional Statements: if – else – elseif, switch. Loops: for – while – break, continue. User-Defined Functions.

3. Arrays, Matrices and Matrix Operations Debugging MATLAB Programs. Working with Data Files, and Graphing Functions: XY Plots – Sub-plots.

4.To solve linear equation

5. Solution of Linear equations for Underdetermined and Over determined cases.

6. Determination of Eigen values and Eigen vectors of a Square matrix.

7. Solution of Difference Equations.

8. Solution of Difference Equations using Euler Method.

9. Solution of differential equation using 4th order Runge- Kutta method.

10. Determination of roots of a polynomial.

11.Determination of polynomial using method of Least Square Curve Fitting.

12.Determination of polynomial fit, analyzing residuals, exponential fit and error bounds from the given data.

13.Determination of time response of an R-L-C circuit

SECTION 2 : Python Programs

1 Introduction to Python Programming:

A. Running instructions in Interactive interpreter and a Python Script.
B. Write a program to purposefully raise Indentation Error and Correct it
C. Write a program to compute distance between two points taking input from the user
D. Write a program add.py that takes 2 numbers as command line arguments and prints its sum.
E. Program to display the following information: Your name, Full Address, Mobile Number, College Name, Course Subjects
F. Write a Program for checking whether the given number is a even number or not.

2 Control Structures, Lists

A. Program to find the largest three integers using if-else
B. Program that receives a series of positive numbers and display the numbers in order and their sum
C. Program to find the product of two matrices and
D. Program to display two random numbers that are to be added, the program should allow the student to enter the answer.
E. If the answer is correct, a message of congratulations should be displayed.
F. If the answer is incorrect, the correct answer should be displayed.
G. Using a for loop, write a program that prints out the decimal equivalents of 1/2, 1/3, 1/4, . 1/10.
H. Write a program using a while loop that asks the user for a number, and prints a countdown from that number to zero.

3 Functions and Recursion

A. Write recursive and non-recursive functions for the following
B. To find GCD of two integers
C. To find the factorial of positive integer
D. To print Fibonacci Sequence up to given number n
E. To display prime number from 2 to n.
F. Function that accepts two arguments: a list and a number n. It displays all of the numbers in the list that are greater than n
G. Functions that accept a string as an argument and return the number of vowels and consonants that the string contains

4 Files, Exceptions, Lists, Sets, Random Numbers

A. Program to write a series of random numbers in a file from 1 to n and display.
B. Program to write the content in a file and display it with a line number followed by a colon
C. Program to display a list of all unique words in a text file
D. Program to analyse the two text files using set operations
E. Write a program to print each line of a file in reverse order.
F. Write a program to count frequency of characters in a given file. Can you use character frequency to tell whether the given file is a Python program file, C program file or a text file?
G. Write a program combine lists that combines these lists into a dictionary.

5 Object Oriented Programming

A. Program to implement the inheritance
B. Program to implement the polymorphism

6 GUI Programming

A. Program that converts temperature from Celsius to Fahrenheit
B. Program that displays your details when a button is clicked
C. Write a GUI for an Expression Calculator using tk

Suggested Readings:

1. Mark Summerfield, "Programming in Python: A Complete Introduction to the Python Language", Addison-Wesley Professional, 2009.
2. Martin C. Brown," PYTHON: The Complete Reference", McGraw-Hill, 2001.
3. W.J. Palm III, Introduction to MATLAB 7 for Engineers, McGraw-Hill International Edition, 2005.
4. Wesley J Chun," Core Python Applications Programming", Prentice Hall, 2012.
5. Allen B Downey," Think Python", O'Reilly, 2012.
6. Stormy Attaway, "MATLAB: A Practical Introduction to Programming and Problem Solving".3rd Edition.

SCHEME OF INSTRUCTION & EXAMINATION
B.E. (Computer Engineering) IV – SEMESTER

S. No.	Course Code	Course Title	Scheme of Instruction				Scheme of Examination			Credits
			L	T	P/D	Contact Hrs/Wk	CIE	SEE	Duration in Hrs	
Theory Courses										
1	HS201EG	Effective Technical Communication in English	3	-	-	3	30	70	3	3
2	HS202MC	Finance and Accounting	3	-	-	3	30	70	3	3
3	PC231CM	OOP using JAVA	3	-	-	3	30	70	3	3
4	PC234CM	Operating System Concepts	3	-	-	3	30	70	3	3
5	PC233CM	Database Management Systems	3	1	-	4	30	70	4	3.5
6	PC232CM	Computer Organization & Microprocessor	3	-	-	3	30	70	3	3
Practical/ Laboratory Courses										
7	PC261CM	Computer Organization & Microporcessor Lab	-	-	3	3	25	50	3	1.5
8	PC262CM	OOP using JAVA Lab	-	-	2	2	25	50	2	1
9	PC263CM	Database Management Systems Lab	-	-	2	2	25	50	2	1
10	PC264CM	Operating System Concepts Lab	-	-	2	2	25	50	2	1
			18	**1**	**09**	**28**	**280**	**620**		**23**

HS: Humanities and Social Sciences BS: Basic Science ES: Engineering Science
MC: Mandatory Course PC: Professional Core
L: Lecture T: Tutorial P: Practical D: Drawing
CIE: Continuous Internal Evaluation SEE: Semester End Evaluation (Univ. Exam)
PO: Political Science, EG: English, CM: Commerce, MT: Mathematics,
CS: Computer Science and Engineering, EC: Electronics and Communication Engineering,

Note:

1. Each contact hour is a clock hour
2. The duration of the practical class is two hours, however it can be extended wherever necessary, to enable the student to complete the experiment.
3. The students have to undergo a Summer Internship of two-week duration after IV – Semester and credits will be awarded in V – Semester after evaluation.

Course Code	Course Title						Core/Elective
HS201EG	**Effective Technical Communication in English**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

To expose the students to:

- Features of technical communication
- Types of professional correspondence
- Techniques of report writing
- Basics of manual writing
- Aspects of data transfer and presentations.

Course Outcomes

On successful completion of the course, the students would be able to:

1. Handle technical communication effectively
2. Use different types of professional correspondence
3. Use various techniques of report writing
4. Acquire adequate skills of manual writing
5. Enhance their skills of information transfer and presentations

UNIT-I

Definition and Features of Technical communication: Definition and features of technical communication (precision, relevance, format, style, use of visual aids), Differences between general writing and technical writing, Types of technical communication (oral and written)

UNIT-II

Technical Writing-I (Official correspondence): Emails, IOM, Business letters, Business proposals.

UNIT-III

Technical writing-II (Reports): Project report, Feasibility report, Progress report, Evaluation report.

UNIT-IV

Technical writing- III (Manuals): Types of manuals, User manual, Product manual, Operations manual.

UNIT-V

Information Transfer and Presentations: Non-verbal (bar diagram, flow chart, pie chart, tree diagram) to verbal (writing), Verbal (written) to non-verbal, Important aspects of oral and visual presentations.

Suggested Readings:

1. Raman, Meenakshi & Sharma, Sangeeta. (2015). *Technical Communication: Principles and Practice* (3rd ed.). New Delhi, OUP.
2. Rizvi, Ashraf, M. (2017). *Effective Technical Communication* (2nd ed.). New Delhi, Tata McGraw Hill Education.
3. Sharma, R. C., & Mohan, Krishna. (2017). *Business Correspondence and Report Writing: A Practical Approach to Business & Technical Communication* (4th ed.). New Delhi, Tata McGraw Hill Education.
4. Tyagi, Kavita & Misra, Padma. (2011). *Advanced Technical Communication*. New Delhi, PHI Learning.
5. Jungk, Dale. (2004). *Applied Writing for Technicians*. New York, McGraw-Hill Higher Education.

Course Code	Course Title						Core/Elective
HS202MC	**Finance and Accounting**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

The course will introduce the students

- To provide basic understanding of Financial and Accounting aspects of a business unit
- To provide understanding of the accounting aspects of business
- To provide understanding of financial statements
- To provide the understanding of financial system
- To provide inputs necessary to evaluate the viability of projects
- To provide the skills necessary to analyse the financial statements

Course Outcomes

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

1. Evaluate the financial performance of the business unit.
2. Take decisions on selection of projects.
3. Take decisions on procurement of finances.
4. Analyse the liquidity, solvency and profitability of the business unit.
5. Evaluate the overall financial functioning of an enterprise.

UNIT-I

Basics of Accounting: Financial Accounting–Definition- Accounting Cycle – Journal - Ledger and Trial Balance-Cash Book-Bank Reconciliation Statement (including Problems)

UNIT-II

Final Accounts: Trading Account-Concept of Gross Profit- Profit and Loss Account-Concept of Net Profit-Balance Sheet (including problems with minor adjustments)

UNIT-III

Financial System and Markets: Financial System-Components-Role-Considerations of the investors and issuers- Role of Financial Intermediaries. Financial Markets-Players- Regulators and instruments - Money Markets Credit Market- Capital Market (Basics only)

UNIT-IV

Basics of Capital Budgeting techniques: Time Value of money- Compounding- Discounting- Future Value of single and multiple flows- Present Value of single and multiple Flows- Present Value of annuities- Financial Appraisal of Projects– Payback Period, ARR- NPV, Benefit Cost Ratio, IRR (simple ratios).

UNIT-V

Financial statement Analysis: Financial Statement Analysis- Importance-Users-Ratio Analysis-liquidity, solvency, turnover and profitability ratios.

Suggested Readings:

1. Satyanarayana. S.V. and Satish. D., Finance and Accounting for Engineering, Pearson Education
2. Rajasekharan, Financial Accounting, Pearson Education
3. Sharma.S.K. and Rachan Sareen, Financial Management, Sultan Chand
4. Jonathan Berk, Fundamentals of Corporate Finance, Pearson Education
5. Sharan, Fundamentals of Financial Management, Pearson Education.

Course Code	Course Title						Core/Elective
PC231CM	**OOP using JAVA**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

- To understand fundamentals of object-oriented programming in Java which includes defining classes, invoking methods, difference between applet and application programs, using class libraries
- To create Java application programs using sound OOP practices such as interfaces, exception handling, multithreading.
- Use Collection framework, AWT and event handling to solve real world problems.
- Exploring Swing, and implementing Servlets.

Course Outcomes

1. Achieve proficiency in object-oriented concepts and also learns to incorporate the same into the Java programming language.
2. Create Java application programs using sound OOP practices e.g. Inheritance, interfaces and proper program structuring by using packages, access control specifiers.
3. Understand and Implement the concepts of Exception Handling in java.
4. Develop the ability to solve real-world problems through software development in high-level programming language using Large APIs of Java as well as the Java standard class library.
5. Understand File, Streams, Input and Output Handling in java.
6. Create graphical user interface and Applets in java as well as apply the knowledge of Event Handling.

UNIT- I

Object Oriented Programming: Principles, Benefits of Object Oriented Programming.

Introduction to Java: Java buzzwords, bytecode. Java Programming Fundamentals: Applet and Application program using simple java program, data types, variables, arrays, operators, expressions, control statements, type conversion and casting, concepts of classes, objects, constructors, methods, access control, this keyword, garbage collection, overloading methods and constructors, introducing access control, static, final, nested and inner classes, exploring string class, using command-line arguments.

Inheritance: Inheritance concept, types of inheritance, Member access rules, use of super and final.

Polymorphism - dynamic binding, method overriding, abstract classes and methods.

UNIT - II

Interfaces: Defining an interface, implementing interfaces, extending interface.

Packages: Defining, Creating and Accessing a Package, importing packages

Exception handling: Benefits of exception handling, classification, checked exceptions and unchecked exceptions, usage of try, catch, throw, throws and finally, rethrowing exceptions, built in exceptions, creating own exception sub classes.

Multithreading: Java Thread Model, The Main Thread, creating a Thread, creating multiple threads, using is Alive() and join(), thread priorities, synchronization, inter thread communication, deadlock.

UNIT- III

Collections: Overview of Java Collection frame work, commonly used Collection classes – Array List, Linked List, Hash Set, Tree Set, Collection Interfaces – Collection, List, Set. Accessing Collection via iterator, working with Map. Legacy classes and interfaces – Vector, Hashtable, Stack, Dictionary, Enumeration interface.

Other Utility classes: String Tokenizer, Date, Calendar, Gregorian Calendar, Scanner

Java Input/Output: exploring java.io, Java I/O classes and interfaces, File, Stream classes, byte stream, character stream, serialization.

UNIT- IV

GUI Programming with java: The AWT class hierarchy, MVC architecture. Applet Revisited: Basics, architecture and skeleton, simple applet program.

Event Handling: Delegation Event Model, Event Classes, Source of Events, Event Listener Interfaces. Handling mouse and keyboard events, Adapter classes.

Database Programming using JDBC: Introduction to JDBC, JDBC Drivers & Architecture, CURD operation Using JDBC, Connecting to non-conventional Databases.

UNIT V

Exploring Swing: JLabel, ImageIcon, JTextField, the Swing buttons, JTabbedpane, JScrollPane, JList, JComboBox.

Servlet: Life cycle, using tomcat, simple servlet, servlet API, javax.servlet package, reading servlet parameters, javax.servlet.http package, handling HTTP requests and responses

Suggested Readings:

1. Herbert Scheldt, "The Complete Reference Java, 7th Edition, Tata McGraw Hill, 2006.

2. James M Slack, Programming and Problem Solving with JAVA, Thomson Learning, 2002.

3. C Thomas Wu, An Introduction to Object Oriented Programming with Java 5th Edition, McGraw Hill Publishing, 2010.

4. H. M. Dietel and P. J. Dietel, Java How to Program, Sixth Edition, Pearson Education / PHI.

Suggested Reference Readings:

1. Understanding OOP with Java, up dated edition, T. Budd, Pearson education.
2. Head First Java, 2nd Edition by Bert Bates, Kathy Sierra Publisher: O'Reilly Media, Inc.
3. An Introduction to programming and OO design using Java, J. Nino and F.A. Hosch, John Wiley & sons.
4. An Introduction to OOP, second edition, T. Budd, Pearson Education.
5. Introduction to Java programming 6th edition, Y. Daniel Liang, Pearson Education.
6. An introduction to Java programming and object oriented application development, R. A. Johnson-Thomas.

Course Code	Course Title						Core/Elective
PC232CM	**Computer Organization & Microprocessor**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

- To understand basic components of computers.
- To explore the I/O organizations in depth.
- To explore the memory organization.
- To understand the basic chip design and organization of 8086 with assembly language.

Course Outcomes: Students will be able to

1. After this course students understand in a better way the I/O and memory organization in depth.
2. Ability to understand the merits and pitfalls in computer performance measurements.
3. Identify the basic elements and functions of 8086 microprocessors.
4. Understand the instruction set of 8086 and use them to write assembly language programs.
5. Demonstrate fundamental understanding on the operation between the microprocessor and its interfacing devices.

UNIT-I

Basic Computer Organization: Functions of CPU, I/O Units, Memory: Instruction: Instruction FormatsOne address, two addresses, zero addresses and three addresses and comparison; addressing modes with numeric examples: Program Control- Status bit conditions, conditional branch instructions, Program Interrupts: Types of Interrupts.

UNIT-II

Input-Output Organizations: I/O Interface, I/O Bus and Interface modules: I/O Vs Memory Bus, Isolated Vs Memory-Mapped I/O, Asynchronous data Transfer- Strobe Control, Hand Shaking: Asynchronous Serial transfer- Asynchronous Communication interface, Modes of transfer Programmed I/O, Interrupt Initiated I/O, DMA; DMA Controller, DMA Transfer, IOP-CPU-IOP Communication, Intel 8089 IOP.

UNIT-III

Memory Organizations: Memory hierarchy, Main Memory, RAM, ROM Chips, Memory Address Map, Memory Connection to CPU, associate memory, Cache Memory, Data Cache, Instruction cache, Miss and Hit ratio, Access time, associative, set associative, mapping, waiting into cache, Introduction to virtual memory.

UNIT-IV

8086 CPU Pin Diagram: Special functions of general purpose registers, Segment register, concept of pipelining, 8086 Flag register, Addressing modes of 8086.

UNIT-V

8086-Instruction formats: assembly Language Programs involving branch & Call instructions, sorting, evaluation of arithmetic expressions.

Suggested Readings:

1. Computer system Architecture: Morris Mano (UNIT-1,2,3).

2. Advanced Micro Processor and Peripherals- Hall/ A K Ray(UNIT-4,5).

3. Computer Organization and Architecture – William Stallings Sixth Edition, Pearson/PHI.

4. Structured Computer Organization – Andrew S. Tanenbaum, 4th Edition PHI/Pearson.

5. Fundamentals or Computer Organization and Design, - Sivaraama Dandamudi Springer Int. Edition.

6. Computer Architecture a quantitative approach, John L. Hennessy and David A. Patterson, Fourth Edition Elsevier.

7. Computer Architecture: Fundamentals and principles of Computer Design, Joseph D. Dumas II, BS Publication.

Course Code	Course Title						Core/Elective
PC234CM	**Operating System Concepts**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives:

- To introduce the concepts of OS structure and process synchronization.
- To study different memory management strategies.
- To familiarize the implementation of file system.
- To understand the principles of system security and protection.
- To discuss the design principles and structure of Windows 7 and Linux

Course Outcomes : Student will be able to

1. Evaluate different process scheduling algorithms.
2. Describe the steps in address translation and different page replacement strategies.
3. Compare different file allocation methods and decide appropriate allocation strategy for given type of file.
4. Explain the mechanisms available in an OS to control access to resource.

UNIT-I

Introduction to Operating Systems: OS structure and strategies, Process concepts, Multithreaded Programming, Process scheduling, Process synchronization, Deadlocks.

UNIT-II

Memory management strategies with example architectures: Swapping, Contiguous allocation, Paging, Segmentation, Segmentation with paging, Virtual memory management: Demand paging, Page replacement, Thrashing.

UNIT-III

File system interface: File concepts, Access methods and protection. File system implementation: File system structure, Allocation methods, Directory implementation of file systems, Mass storage structures, I/O systems.

UNIT-IV

System Protection: Principles and Domain, Access Matrix and implementation, Access control and access rights, Capability based systems, and Language based Protection System Security: Problem, Program threats, cryptography, user authentication, implementing security defenses, Firewalling, Computer Security Classification.

UNIT-V

Case Studies: The Linux System–Design principles, Kernel modules, Process management, Scheduling, Memory management, File systems, Input and Output, Inter process communication. Windows7–Design principles, System components, Terminal services and fast user switching File systems, Networking, Programmer interface.

Suggested Reading:

1. Abraham Silberschatz, Peter B Galvin, Operating System Concepts,9th Edition, Wiley, 2016
2. William Stallings, Operating Systems-Internals and Design Principles, 8thedition, Pearson, 2014
3. Andrew S Tanenbaum, Modern Operating Systems, 4th edition, Pearson, 2016.

Course Code	Course Title						Core/Elective
PC233CM	**Database Management Systems**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**3**	**-**	**-**	**-**	**30**	**70**	**3**

Course Objectives

- To Learn mathematical concepts as applied in computer
- To introduce three scheme architecture and DBMS functional components.
- To learn formal and commercial query languages of RDBMS
- To Study different file organization and indexing techniques
- To familiarize theory of serializablity and implementation of concurrency control, and recovery

Course Outcomes

1. Understand the mathematical foundations on which RDBMS are built
2. Model a set of requirements using the Extended Entity Relationship Model (EER), transform an EER model into a relational model and refine the relational model using theory of normalization
3. Develop Database application using SQL and Embedded SQL
4. Use the knowledge of file organization and indexing to improve database application performance
5. Understand the working of concurrency control and recovery mechanisms in RDBMS

UNIT-I

Introduction: Database System Application, Purpose of Database Systems, View of Values, Nested Sub-queries, Complex Queries views, Modification of the Database, Joined Relations

Data, Database Language, Relational Databases, Database Design, Object-Based and Semi-Structured Databases, Data Storages and Querying , Transaction Management ,Data Mining and Analysis, Database Architecture, Database Users and Administrators.

Database Design and the E-R Model: Overview of the Design Process, The Entity Relationship Model Constraints, Entity-Relationship Design issues, Weak Entity Sets Extended E-R Features Database Design for banking Enterprise, Reduction to Relational Schemas, Other Aspects of Database Design

UNIT-II

Relational Model: Structure of Relational Databases, Fundamental Relational-Algebra Operations, Additional Relational-Algebra Operations, Extended Relational-Algebra Operations, Null Values, Modification of the Databases

Structured Query Language: Data Definition, Basic Structure of SQL Queries, Set Operations, Aggregate Functions, Null

UNIT-III

Advanced SQL: SQL Data Types and Schemes, Integrity constraints, Authorization, Embedded SQL, Dynamic SQL, Functions and Procedural Constructs, Recursive Queries, Advanced SQL Features.

Relational Database Design: Features of Good Relational Design, Atomic Domains and First Normal Form, Functional Dependency Theory, Decomposition using Functional Dependencies.

UNIT-IV

Indexing and Hashing: Basic Concepts, Ordered Indices, B*-tree index files, B-tree index files, multiple key access, static hashing, dynamic hashing, comparison of ordered indexing and hashing bitmap indices.
Index definition in SQL transactions: Transaction concepts, transaction state, implementation of atomicity and durability, concurrent executions, serializability, recoverability, implementation of isolation, testing for serializability.

UNIT-V

Concurrency Control: Lock based protocols, timestamp based protocols, validation based protocols, multiple granularity, multi version schemes, deadlock handling, insert and delete operations, weak levels of consistency, concurrency of index structures.
Recovery system: Failure classification, storage structure, recovery and atomicity, log-based recovery, recovery with concurrent transactions, buffer management, failure with loss of non-volatile storage, advanced recovery techniques, remote backup systems.

Suggested Readings:

1. Abraham Silberschatz, Henry F Korth, S Sudarshan, Database System Concepts, McGraw-Hill, 6th Edition, 2010
2. Ramakrishnan, Gehrke, Database Management Systems, McGraw-Hill, 3rd Edition, 2003
3. Elmasri, Navathe, Somayajulu, Fundamentals of Database Systems, Pearson Education, 4th Edition, 2004.

Course Code	Course Title						Core/Elective
PC261CM	**Computer Organization & Microprocessor Lab**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**-**	**-**	**-**	**2**	**25**	**50**	**1**

Course Objectives

The objectives of the course are to impart knowledge of the:

- To become familiar with the architecture and Instruction set of Intel 8086 microprocessor.
- To provide practical hands on experience with Assembly Language Programming.
- To familiarize the students with interfacing of various peripheral devices with 8085 microprocessors.

Course Outcomes

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

1. Interpret the principles of Assembly Language Programming, instruction set in developing microprocessor based applications.
2. Develop Applications such as: 8-bit Addition, Multiplication, Division, array operations, swapping, negative and positive numbers.
3. Analyse the interfaces like serial ports, digital-to-analog Converters and analog-to-digital converters etc.
4. Build interfaces of Input-output and other units like stepper motor with 8086.
5. Analyse the function of traffic light controller.

List of Experiments:

1. Tutorials with 8086 kit / MASM software tool.
2. Fixed-point multiplication and division.
3. Floating-point multiplication and division.
4. Sorting hexadecimal array.
5. Code conversion from hexadecimal to decimal.
6. Sum of set of BCD numbers.
7. Searching.
8. Display a string of characters using 8279.
9. Interfacing traffic light controller using 8255.
10. Interfacing seven-segment LED using 8255.
11. Interfacing stepper motor using 8255.
12. Interfacing 8253 counter.
13. D/A conversion using 8255.
14. A/D conversion using 8255.

Suggested Readings:

1. Yu-cheng Liu, Glenn A. Gibson, "Microcomputer Systems: The 8086/8088 Family", 2nd Edition, PHI Learning 2011.

2. Douglas Hall. "Microprocessor and Interfacing programming and Hardware", Tata Mc Graw Hill, Revised 2nd Edition, 2007.

3. Brey B. Brey, "The Intel Microprocessor, 8086/8088, 80186/80188, 80286, 80386, 80486, Pentium and Pentium Pro Processors-Architecture, Programming and interfacing", 4th Edition, Prentice Hall, 1993.

Course Code	Course Title						Core/Elective
PC264CM	**Operating Systems Lab**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	-	-	-	**2**	**25**	**50**	**1**

Course Objectives

- To learn shell programming and the use of filters in the LINUX environment.
- To practice multithreaded programming.
- To implement CPU Scheduling Algorithms and memory management algorithms.

Course Outcomes

After completing this course, the student will be able to:

1. Write shell scripts for simple system administration tasks.
2. Write concurrent programs with synchronization constricts.
3. Compare the performance of various CPU Scheduling Algorithm.
4. Critically analyze the performance of the various Memory management algorithms

List of Experiments:

1-3. Memory Management Algorithms
4-5. Examples of Multithreading
6. Producer & Consumer problem using Semaphores and shared memory
7-8. Processor Scheduling algorithms
9. Dining Philosophers problem using Semaphores
10. Readers and Writers problem using Semaphores
11. Shell-programming exercises.

Course Code	Course Title						Core/Elective
PC263CM	**Database Management Systems Lab**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**-**	**-**	**-**	**2**	**25**	**50**	**1**

Course Objectives

- To practice various DDL commands in SQL
- To write simple and complex queries in SQL
- To familiarize PL/SQL

Course Outcomes

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

1. Design and implement a database schema for a given problem
2. Populate and query a database using SQL and PL/SQL
3. Develop multi-user database application using locks

Creation of database (exercising the commands for creation)

1. Simple to complex condition query creation using SQL Plus.
2. Usage of triggers and stored procedures
3. Creation of forms for student information, library information, pay roll etc.
4. Writing PL/SQL procedures for data validation.
5. Report generation using SQL reports.
6. Creating password and security features for applications.
7. Using of file locking, table locking facilities in applications.
8. Creation of small full-fledged database application spreading over 3 sessions.

Note: The creation of sample database for the purpose of the experiments is expected to be pre-decided by the instructor.

Course Code	Course Title						Core/Elective
PC262CM	**OOP using JAVA Lab**						**Core**
Prerequisite	Contact Hours per Week				CIE	SEE	Credits
	L	T	D	P			
-	**-**	**-**	**-**	**2**	**25**	**50**	**1**

Course Objectives

- Ability to learn the concept of classes, inheritance and abstract classes.
- To Learn to demonstrate multithreaded programs with synchronization.
- To Demonstrate real world applications using java collection frame work and I/O classes.
- To Model Event driven GUI programs using AWT/Swing.
- To build software development skills using java programming for real world applications.
- To implement frontend and backend of an application**.**
- To implement classical problems using java programming.

Course Outcomes

After completing this course, the student will be able to:

1. Able to understand the OOPS features.
2. Implement the concepts of Exception Handling in java Applications.
3. Read and write data using different Java I/O streams.
4. Create graphical user interfaces and Applets by applying the knowledge of Event Handling.
5. Create robust applications using Java standard class libraries and retrieve data from a database with JDBC.
6. Ability to solve real-world problems by designing user friendly GUI with befitting backend through the APIs of Java.

List of Experiments:

1) Write a Java program to illustrate the concept of class with method overloading.
2) Write a Java Program that reads a line of integers, and then displays each integer, and the sum of all the integers (Use String Tokenizer class of java. util)
3) Write a Java program to illustrate the concept of Single level and Multi level Inheritance.
4) Write a Java program to demonstrate the Interfaces & Abstract Classes.
5) Write a Java program to implement the concept of exception handling.
6) Write a Java program to illustrate the concept of threading using Thread Class and runnable Interface.
7) Write a Java program to illustrate the concept of Thread synchronization.
8) Write a Java program that correctly implements producer consumer problem using the concept of inter thread communication.
9) Write a Java program to illustrate collection classes like Array List, LinkedList, Tree map and Hash map.
10) Write a Java program to illustrate Legacy classes like Vector, Hashtable, Dictionary & Enumeration interface
11) Write a Java program to implement iteration over Collection using Iterator interface and List Iterator interface
12) Write a Java program that reads a file name from the user, and then displays information about whether the file exists, whether the file is readable, whether the file is writable, the type of file and the length of the file in bytes.
13) Write a Java program to illustrate the concept of I/O Streams

14) Write a Java program to implement serialization concept
15) Write a Java applet program to implement Colour and Graphics class
16) Write a Java applet program for handling mouse & key events
17) Write a Java applet program to implement Adapter classes
18) Write a Java program that works as a simple calculator. Use a grid layout to arrange buttons for the digits and for the +, -, *, % operations. Add a text field to display the result.
19) Write an example for JDBC prepared statement with Result Set.
20) Program to get primary key value (auto-generated keys) from inserted queries using JDBC
21) Program to create a simple JList
22) java Program to create a simple checkbox using JCheckBox
23) Program to create a checkbox and ItemListener to it.
24) 1. Write Servlet application to print current date & time
    2. Html & Servlet Communication
    3. Auto refresh a page
    4. Demonstrate session tracking
    5. Select record from database
    6. Application for login page
    7. Insert record into database
    8. Count the visits on web page
    9. Insert teacher record in Database