CHAPTER – 2
SEMESTER-WISE STRUCTURE OF CURRICULUM
Mandatory Induction Program

3 Weeks Duration
• Physical activity • Creative Arts • Universal Human Values • Literary • Proficiency Modules • Lectures by Eminent people • Visit to local areas • Familiarization of Dept./Branch Innovations

ENGINEERING FIRST YEAR: SEMESTER-1							
Sl. No.	**Course Type**	**Course Code**	**Course Title**	**Hours per week**			**Credits**
				L	**T**	**P**	
1	BSC	20MA1102	Calculus & Linear Algebra	3	1	0	4
2	ESC	20EE1109	Basic Electrical and Electronics Engg.	3	1	0	4
3	ESC	20CS1101	Problem Solving and Programming Through C	3	1	0	4
4	ESC	20ME1114	Engineering Graphics & Computer Drafting	1	0	3	2.5
5	HSC	20EG1181	English-Language communication Skills Lab-I	~~0~~ 1	0	3	2.5
6	ESC	20EE1189	Basic Electrical and Electronics Engg. Lab	0	0	3	1.5
7	ESC	20CS1181	Problem Solving and Programming Through C Lab	0	0	3	1.5
8	MC	20HS1101	Indian Constitution	2	0	0	0
Total				**13**	**3**	**12**	**20**

ENGINEERING FIRST YEAR:SEMESTER-2							
Sl. No.	**Course Type**	**Course Code**	**Course Title**	**Hours per week**			Credits
				L	**T**	**P**	
1	BSC	20MA1202	Discrete Mathematics	3	1	0	4
2	BSC	20PY1201	Engineering Physics	3	1	0	4

3	HSC	20BM1201	Managerial Economics and Finance Analysis	3	0	0	3
4	PCC	20CS1201	Object Oriented Programming through Java	3	1	0	4
5	PCC	20CS1202	Data Structures	3	0	0	3
6	BSC	20PY1281	Engineering Physics Lab	0	0	3	1.5
7	PCC	20CS1281	Object Oriented Programming through Java Lab	0	0	3	1.5
8	PCC	20CS1282	Data Structures Lab	0	0	3	1.5
9	HSC	20BE1201	Environmental Science	2	0	0	0
Total				**17**	**3**	**9**	**22.5**

ENGINEERING SECOND YEAR: SEMESTER-1							
Sl. No.	**Course Type**	**Course Code**	**Course Title**	**Hours per week**			**Credits**
				L	**T**	**P**	
1	BSC	20MA2102	Probability and Statistics	3	1	0	4
2	ESC	20EC2110	Digital Logic Design	3	0	0	3
3	PCC	20CS2101	Design & Analysis of Algorithms	3	1	0	4
4	PCC	20CS2102	Database Management Systems	3	0	0	3
5	PCC	20CS2103	Formal Languages & Automata Theory	3	0	0	3
6	PCC	20CS2181	Design & Analysis of Algorithms Lab	0	0	3	1.5
7	ESC	20EC2180	Digital Logic Design Lab	0	0	3	1.5
8	PCC	20CS2182	Database Management Systems Lab	0	0	3	1.5
Total				**15**	**2**	**9**	**21.5**

ENGINEERING SECOND YEAR:SEMESTER-2							
Sl. No.	**Course Type**	**Course Code**	**Course Title**	**Hours per week**			**Credits**
				L	**T**	**P**	
1	HSC	20BM2202	Introduction to Operation Research	3	0	0	3
2	PCC	20CS2201	Computer Organization & Architecture	3	0	0	3
3	PCC	20CS2202	Data Science with Python	3	0	0	3
4	PCC	20CS2203	Web Technologies	3	0	0	3
5	PCC	20CS2204	Compiler Design	3	0	0	3
6	PCC	20CS2281	Computer Organization & Architecture Lab	0	0	3	1.5
7	PCC	20CS2282	Data Science with Python Lab	0	0	3	1.5
8	PCC	20CS2283	Web Technologies Lab	0	0	3	1.5
Total				**15**	**0**	**9**	**19.5**

ENGINEERING THIRD YEAR:SEMESTER-1							
Sl. No.	**Course Type**	**Course Code**	**Course Title**	**Hours per week**			**Credits**
				L	**T**	**P**	
1	PCC	20CS3101	Operating System	3	0	0	3
2	PCC	20CS3102	Computer Networks	3	0	0	3
3	PCC	20CS3103	Software Engineering	3	0	0	3
4	PCC	20CS3104	Mathematical Foundations for Data Science	3	0	0	3
5	PEC	20CS31XX	Elective – I	3	0	0	3
6	PCC	20CS3181	Operating System Lab	0	0	3	1.5
7	PCC	20CS3182	Computer Networks Lab	0	0	3	1.5
8	PCC	20CS3183	Software Engineering Lab	0	0	3	1.5
9	HSC	20EG3182	English-Language communication Skills Lab- II	0	0	3	1.5
Total				**15**	0	12	21

ENGINEERING THIRD YEAR:SEMESTER-2							
Sl. No.	**Course Type**	**Course Code**	**Course Title**	**Hours per week**			**Credits**
				L	**T**	**P**	
1	PCC	20CS3201	Cryptography and Networks Security	3	1	0	4
2	PCC	20CS3202	Artificial Intelligence	3	1	0	4
3	PEC	20CS32XX	Elective – II	3	0	2	3
4	PEC	20CS32XX	Elective – III	3	0	2	3
5	OEC	20XX32XX	Open Elective-I	3	0	0	3
6	HSC	EG3283	English-Language communication Skills Lab-I -III	0	0	3	1.5

7	PR	20CS3291	Mini Project	0	0	6	3
8	MC	20CS3203	Career Development Course	2	0	0	0
9		20CS3292	Summer Internship	0	0	6	3
Total				**16**	**0**	**15**	**21.5**
10	20CS3292		Summer Internship	0	0	6	3

ENGINEERING FOURTH YEAR:SEMESTER-1							
Sl. No.	**Course Type**	**Course Code**	**Course Title**	**Hours per week**			**Credits**
				L	**T**	**P**	
1	PCC	20CS4101	Machine Learning	3	1	0	4
2	PEC	20CS41XX	Elective-IV	3	0	0	3
3	OEC	20XX41XX	Open Elective – II	3	0	0	3
4	PR	20CS4193	Project-I	0	0	12	6
Total				**9**	**1**	**12**	**16**

ENGINEERING FOURTH YEAR:SEMESTER-2							
Sl. No.	**Course Type**	**Course Code**	**Course Title**	**Hours per week**			**Credits**
				L	**T**	**P**	
1	PEC	20CS42XX	Elective-V	3	0	0	3
2	OEC	20XX42XX	Open Elective-III	3	0	0	3
3	OEC	20CS42XX	Open Elective-IV	3	0	0	3
4	PR	20CS4294	Project-II	0	0	12	6
5	HSC	20CS4299	Community Service	0	0	4	2
Total				**9**	**0**	**16**	**17**

LIST OF PROFESSIONAL ELECTIVE COURSES

PROGRAM ELECTIVE COURSES							
Sl. No.	**Course Type**	**Course Code**	**Course Title**	**Hours per week**			**Credits**
				L	**T**	**P**	
1	PEC	20CS3121	Data Mining	3	0	0	3
2	PEC	20CS3122	Mobile Application Development	3	0	0	3
3	PEC	20CS3123	Distributed Computing	3	0	0	3
4	PEC	20CS3124	Advanced Computer Architecture	3	0	0	3
5	PEC	20CS3221	Object Oriented Analysis & Design (OOAD)	3	0	**0**	3
6	PEC	20CS3123	Distributed Computing	3	0	0	3
7	PEC	20CS3223	Real Time Operating System	3	0	0	3
7	PEC	20CS3223	Embedded Systems	3	0	0	3
8	PEC	20CS3225	Digital Image Processing	3	0	0	3
9	PEC	20CS3231	Information Retrieval	3	0	0	3

10	PEC	20CS3232	Software Testing	3	0	0	3
11	PEC	20CS3233	Mobile Computing	3	0	0	3
12	PEC	20CS3234	Data Compression	3	0	0	3
13	PEC	20CS3235	Computer Graphics	3	0	0	3
14	PEC	20CS4141	Data Science	3	0	0	3
15	PEC	20CS4142	Unix and Shell Programming	3	0	0	3
16	PEC	20CS4143	VLSI	3	0	0	3
17	PEC	20CS4144	Soft Computing	3	0	0	3
18	PEC	20CS4145	File Structure	3	0	0	3
19	PEC	20CS4251	Optimization Technique	3	0	0	3
20	PEC	20CS4252	Design Patterns	3	0	0	3
21	PEC	20CS4253	Cloud Computing	3	0	0	3
22	PEC	20CS4254	Block Chain Technology	3	0	0	3
23	PEC	20CS4255	Internet Of Things	3	0	0	3
24	PEC	20CS4257	Computer Vision	3	0	0	3

LIST OF OPEN ELECTIVE COURSES OFFERED BY DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

OPEN ELECTIVE COURSES FOR ALL BRANCHES							
Sl. No.	**Course Type**	**Course Code**	**Course Title**	**Hours per week**			**Credits**
				L	**T**	**P**	
1	OEC	20CS4261	Big Data Analytics	3	0	0	3
2	OEC	20CS4262	Biometric Security	3	0	0	3
3	OEC	20CS4263	Human Computer Interaction	3	0	0	3
4	OEC	20CS4264	Cyber Security	3	0	0	3
5	OEC	20CS4265	Robotics	3	0	0	3
6	OEC	20CS4266	Computer Forensics	3	0	0	3
OPEN ELECTIVE COURSES FOR ALL BRANCHES except CSE							
7	OEC	20CSXX71	Object Oriented Programming through Java	3	0	0	3
8	OEC	20CSXX72	Database Management System	3	0	0	3
9	OEC	20CSXX73	Computer Graphics	3	0	0	3
10	OEC	20CSXX74	Distributed Computing	3	0	0	3
11	OEC	20CSXX75	Digital Image Processing	3	0	0	3

LIST OF COURSES OFFERED TO OTHER ENGINEERING BRANCHES BY DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

COURSES for other Engg. Branches							
Sl. No.	Course Type	Course Code	Course Title	Hours per week			Credits
				L	T	P	
1	ESC	20CSXX09 (ECE)	Object Oriented programming	2	0	0	2
2	ESC	20CSXX89 (ECE)	Object Oriented programming lab	0	0	3	1.5
3	ESC	20CSXX10 (ECE)	Computer Organization and Architecture	3	1	0	4
4	ESC	20CSXX08 (all branches except CSE)	Programming & Data Structures	3	0	0	3
5	ESC	20CSXX88 (all branches except CSE)	Programming & Data Structures lab	0	0	3	1.5
6	ESC	20CSXX11 (ECE)	Computer Networks	3	0	0	3
7	ESC	20CSXX07 (CHE)	Object Oriented programming through JAVA	3	0	0	3
8	ESC	20CSXX87 (CHE)	Object Oriented programming through JAVA lab	0	0	3	1.5

****
****
****

ENGINEERING FIRST YEAR: SEMESTER-I

****

Course code	Course Name	Course Category	L-T-P	Credits
20MA1102	**Calculus and Linear Algebra (CSE)**	**BSC**	**3-1-0**	**4**

Course Learning Objectives: The objective of this course is to
1. Discuss the Solutions of first order differential equations
2. Understand Continuity and differentiability of multi-variable functions and its applications to discuss maximum and minimum
4.Discuss the linear transformation and its Eigen values and Eigen vectors.
5. Discuss numerical methods to find the roots of polynomial and transcendental equations Interpolating and Fitting the curves for data points.
6. Evaluate integrals by using numerical methods and solving IVP

Unit – I **(10 Contact hours)**

Differential equations of first order and first degree:

Basic concepts, Variable Separable method, homogeneous differential equations, Exact differential equations, Integrating factor, Differentiable equations Reducible to exact, Linear differential equations, Bernoulli differential equations

Unit - II **(12 Contact hours)**

Functions of several variables:

Limit, Continuity and Differentiability of functions of several variables, Partial derivatives and their geometrical interpretation, Differentials, Derivatives of Composite and Implicit functions, Chain rule, Jacobians, Derivatives of higher order, Homogeneous functions, Euler's theorem, and Harmonic functions.

Unit - III

Applications of Functions of several Variable: **(8 Contact hours)**

Taylor's expansion of functions of several variables, Maxima and Minima of functions of several variables - Lagrange's method of multipliers.

Unit – I V

Linear Algebra: **(10 Contact hours)**

Vector Spaces, Linear Combinations of Vectors, Linear dependence and Independence, Basis and Dimension, Linear Transformations, Matrix Representations of Linear transformation,

Unit-V **(10 Contact hours)**

Matrix Algebra (Eigen Values and Eigen Vectors):

Solving system of Homogeneous and Non-Homogeneous equations by using Gauss elimination method. Characteristic roots and Characteristic Vectors of a matrix - Cayley-Hamilton Theorem (without proof); Finding inverse and power of a matrix by Cayley-Hamilton Theorem.

Unit - VI **(10 Contact hours)**

Numerical solution of transcendental equations, Interpolation:

Roots of polynomial and transcendental equations – bisection method, Regula-falsi method and Newton-Raphson Method, Finite differences, Newton's forward and backward interpolation formulae, Gauss central difference Interpolation formulae.

Learning resources

Text book:

1. ERWIN KREYSZIG, '*Advanced Engineering Mathematics'*, Wiley-India,9th Edition

Reference Books:

1. TOM M. APOSTAL,*'Calculus, Volume II'*, Wiley-India,Second Edition,
2. R. K. JAIN AND S. R. K. IYENGAR,'*Advanced Engineering Mathematics'*, Narosa Publishers,3rd Edition.
3. B.S.GREWAL, '*Higher Engineering Mathematics'*, Khanna Publishers,42nd Edition.

Web resources:

1. NPTEL, IIT- Madras,08-June-2017, Introduction to ordinary differential equations URL: https://nptel.ac.in/courses/111106100/12

2. NPTEL, IIT- Kanpur,15-March-2016,Differential Calculus of Several Variables URL:https://nptel.ac.in/courses/111104092/11

3.NPTEL, IIT- Madras,2015-02-05,Linear Algebra URL: https://nptel.ac.in/courses/111/106/111106051/

4.NPTEL, IIT- Delhi, 2009-12-31,Numerical Methods and Computation, URL: https://nptel.ac.in/courses/122/102/122102009/

5. NPTEL, IIT- Kharagpur, 2012-07-11,Regression Analysis,URL: https://nptel.ac.in/courses/111/105/111105042/

6.RGUKT Course Content

Course outcomes: At the end of the course, the student will be able to

CO 1	Solve first order differential equations.
CO 2	Explain limits and continuity, differentiability and partial derivatives of functions of multivariable and solve the extremum problems subjected to constraints.
CO 3	Find Taylor's series and extreme values for functions of two variables .
CO 4	Finding Eigen values and Eigen vector for a linear transformation.
CO 5	Approximate the roots of polynomial and transcendental equations.
CO 6	Approximate the value at a point by using given discrete data. Solve IVP numerically.

For Theory courses only:

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

****

Course Code	Course Name	Course Category	L-T-P	Credits
20EE1109	Basic Electrical and Electronics Engineering	ESC	3-1-0	4

Course Learning Objectives

1. Impart a basic knowledge of electrical quantities such as current, voltage, power, energy and frequency to understand the impact of technology.
2. Provide knowledge for the analysis of basic DC and AC circuits used in electrical and electronic devices.
3. To explain the working principle, construction, applications of DC machines, AC machines.
4. Realize the importance of electronic devices in the present technology.

Course Content:

Unit-I **(13 Contact hours)**

DC Circuits: Parallel circuits, Star-delta and delta-star transformations, equivalent resistance calculation, Mesh and Nodal analysis, superposition theorem, thevenin's theorem and maximum

power transfer theorem. Introduction, Basic definitions, Types of elements, Ohm's Law, Kirchhoff's Laws, Series.

Unit-II (**7 Contact hours)**

AC Circuits

Single-phase: Inductive circuits, capacitive circuits, series RL, RC and RLC circuits, resonance
Three-phase: star connection and delta connection.

Unit-III **(9 Contact hours)**

DC Machines

Generator: Principle of operation of DC Generator, EMF equation, types, applications **Motor:** DC motor types, torque equation, applications, three point starter.

UNIT-IV **(9 Contact hours)**

AC Machines

Transformers: Principle of operation of single phase transformers, EMF equation, losses, efficiency and regulation.
Induction Machine: Principle of operation of induction motor, slip-torque characteristics, applications.

UNIT-V **(9 Contact hours)**

Semiconductor Devices

Diode: types of semiconductors, P-N junction diode, V-I Characteristics, zener diode, Diode Applications. **Rectifiers:** Half wave, Full wave and Bridge rectifiers.

UNIT-VI **(7 Contact hours)**

Transistors

PNP and NPN Junction transistor, Transistor configurations, Transistor as an amplifier

Learning Resources

Text books:

Basic Electrical and Electronics Engineering by Kothari and Nagarath, TMH Publications, 2nd Edition.

Reference Books:

Principles of Electrical and Electronics Engineering by V.K.Mehta, S.Chand & Co.
Basic Electrical Engineering by Kothari and Nagarath, TMH Publications, 2nd Edition.

Web Resources:

1. Prof T S Natarajan, NPTEL-IIT Madras, 'Basic Electronics'
   URL: https://nptel.ac.in/courses/122106025/
2. Prof U Umanand, IISC Bangalore, 'Basic Electrical Technology'.
   URL: http://nptel.ac.in/courses/108108076/
3. Prof S Aniruddhan, IIT Madras, 'Basic Electrical Circuits'.
   URL: https://onlinecourses.nptel.ac.in/noc16_ee03

Course Outcomes:

At the end of the course, the student will be able to

CO 1	Predict the behaviour of any electrical and magnetic circuits.
CO 2	Formulate and solve complex AC, DC circuits
CO 3	Identify the type of electrical machine used for that particular application
CO 4	Realize the requirement of transformers in transmission and distribution of electric power and other applications

CO 5	Utilize the semiconductor devices like diodes and transistors
CO 6	Internlink Knowledge of electrical and electronic circuits to general problems

Assessment Method:

Assessment Tool	Weekly tests/Assignments (In semester)	Monthly tests (In semester)	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

****

Course Code	Course Name	Course Category	L-T-P	Credits
20CS1101	**Problem Solving and Programming Through C**	**ESC**	**3-1-0**	**4**

Course Learning Objectives:

1. To understand the various steps in Program development.
2. To understand the basic concepts in C Programming Language.
3. To learn about arrays and character arrays
4. To learn how to write modular and readable C Programs
5. To understand the basic concepts of Pointers and Dynamic memory allocation.
6. To understand the usage of Structure and Unions and about file operations

Course Content:

UNIT – I **(10 Contact hours)**

Introduction to Computer Programming: Computing Environments, Computer Languages, Creating and Running Programs. Algorithms and Flow charts :

Definition of Algorithms and examples, Introduction to C Language - Background, C Identifiers, Data Types, Operators, Variables, Constants, Input / Output, Expressions, C Programs, Precedence and Associativity, Evaluating Expressions, Type Conversion, Statements, Bitwise Operators.

UNIT-II **(10 Contact hours)**

Conditional Statements and Loops: if-else, switch Statements, Standard Functions. Repetition: loops, while, for, do-while statements, Loop examples, break, continue and GOTO statements.

UNIT-III **(8 Contact hours)**

Arrays: Array Concepts, Using Arrays in C, Array Applications, Two- Dimensional Arrays, Multidimensional Arrays;

Strings: Declaring, Initializing, Printing and reading strings, string manipulation functions, String input and output functions, array of strings,

UNIT – IV **(12 Contact hours)**

Functions: Designing Structured Programs, Function Basics, User Defined Functions, Inter Function Communication, Standard Functions, Recursion- Recursive Functions, Preprocessor Commands. Strings - Concepts, C Strings, String Input / Output Functions, Arrays of Strings, String Manipulation Functions.

UNIT – V **(10 Contact hours)**

Pointers: Introduction, Pointers to Pointers, Compatibility, void Pointers, Arrays and Pointers, Pointer constants, Pointers and Strings, Pointers to Functions, Pointers to Constant Objects, Constant Pointers, Pointer Arithmetic. Call-by-reference: Pointers for Inter-Function Communication, Passing Arrays to a Function.

Dynamic Memory Allocation: Memory Allocation Functions, Programming Applications, Command-line Arguments.

UNIT – VI **(10 Contact hours)**

The Type Definition (type def), Enumerated Types .

Structure& Union: Definition and Initialization of Structures, Accessing Structures, Nested Structures, Arrays of Structures, Structures and Functions, Pointers to Structures, Self-Referential Structures, definition and Initialization of Union, Accessing of Union.

Files: Input and Output: Files, Streams, Standard library Input Output Functions, Character Input Output Functions.

Learning resources

Text book:

1. Reema Thareja, " Programming in C", Oxford Publications, 2nd Edition

Reference Books:

1. E. BalaguruSwamy, " Programming in ANSI C", Mc Graw Hill, 7th Edition
2. Brian W. Kernighan, Dennis M. Ritchie, " The C Programming Language", Prentice Hall, 2nd Edition

Web resources:

Indian Institute of Technology, Kharagpur, "Problem Solving through Programming in C", https://nptel.ac.in/courses/106105171/

Course outcomes: At the end of the course, the student will be able to

CO 1	Develop flowcharts, algorithms for given complex problems.
CO 2	Analyze basic programming constructs.
CO 3	Write C programs for real world problems. Implement C programming by using various control structures.
CO 4	Able to write rite modular and readable C Programs
CO 5	Able to use pointers in C programming
CO 6	Appreciate coding standards and best practices for program development.

For Theory courses only:

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

****

Course Code	**Course Name**	**Course Category**	**L-T-P**	**Credits**
ME1214	**Engineering Graphics & Computer Drafting**	**ESC**	**1-0-3**	**2.5**

Course Learning Objectives:

1. To know about emergence of Engineering Graphics as a refined communication tool and to be aware of International and national standards of practice for uniform presentation of drawings.
2. To adopt the projection of three-dimensional object orthogonally on a set of vertical and horizontal planes and obtain the views of the frontal and the top surfaces.
3. To describe the position of a point and position of the line with respect to all the planes of projection and obtain its views.

4. To learn orthographic projections of various simple plane surfaces in simple and inclined positions.
5. To know about orthographic projections of right and regular solids in simple positions, when their axes are perpendicular to one reference plane and parallel to the other.
6. To learn about types of cutting planes and to obtain views of simple solids.
7. To learn about different methodologies to be used for obtaining the two dimensional layout of the lateral surfaces of uncut solids.
8. To learn about computer aided drafting techniques and to be familiarize with one of the most powerful software 'AutoCAD'.

Course Content:

Unit - I **(6 Contact hours)**

Introduction to Engineering drawing – Tools and Standards, Geometric Constructions, Scales, Conics and Special Curves - ellipse, parabola, hyperbola, cycloids, Involutes.

Unit - II **(6 Contact hours)**

Introduction to Orthographic Projections,Projections of Points, Projection of Lines.

Unit - III **(7 Contact hours)**

Projection of Planes, Projections of Solids cube, prism, pyramid, cylinder, cone and sphere.

Unit - IV **(5 Contact hours)**

Sections of Solids - cube, prism, pyramid, cylinder, cone and sphere. Development of Surfaces – Parallel line method and Radial line method.

Unit - V **(6 Contact hours)**

Computer Aided Design – Introduction to AutoCAD, Co-ordinate System (UCS) and their Commands, Basic Commands of Drawing and Editing, Dimensioning and Text.

Unit – VI **(6Contact hours)**

Drawing practice with AutoCAD – Creating 2D Drawings of Objects from Isometric views, Creating Isometric views form Orthographic views and . Introductions to 3D drawings.

Learning Resources

Text Book:

1. N.D. Bhatt and V.M. Panchal, "Engineering Drawing", Charotar Publications

Reference Books:

1. K. Venugopal , "Engineering drawing"
2. N.S. Parthasarathy and Vela Murali , "Engineering Drawing"

Course outcomes: At the end of the course, the student will be able to

CO 1	Student will be aware of International and national standards of practice.
CO 2	Student will be familiar with obtaining the views of the frontal and the top surfaces of an object.
CO 3	Student will be aware of orthographic projections of right and regular solids in simple positions, when their axes are perpendicular to one reference plane and parallel to the other.
CO 4	Student will know about computer aided drafting techniques and will be familiar with one of the most powerful software 'AutoCAD'.

For Theory Courses only:

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Practice charts (Average of	Monthly tests	End Semester Test	Total

	min 8 charts)			
Weightage (%)	20%	20%	60%	100%

****

Year & Sem: E1 & SEM1	Course Code: 20EG1181	Course Name: ENGLISH LANGUAGE COMMUNICATION SKILLS LAB-I	L – T -P: 1 – 0 – 3	Credits 2.5

Course objectives:

1. To facilitate computer-aided multi-media instruction enabling individualized and independent language learning
2. To sensitize the students to the nuances of English speech sounds, word accent, intonation and rhythm
3. To provide opportunities for practice in using English in day to day situations
4. To improve the fluency in spoken English and neutralize mother tongue influence
5. To train students to use language appropriately for debate, group discussion and public speaking

COURSE CONTENT

UNIT-I: **(06 Contact Hours)**

Theory: An Ideal Family by Katherine Mansfield

Spoken Skills: Situational Dialogues – Role-play – Expressions in various situations – Self Introduction – Introducing others – Greetings – Apologies – Requests – Giving directions

UNIT-II: **(06 Contact Hours)**

Theory: Energy -Alternative sources of Energy

Panel Debate on "On-grid & off-grid support to public participation in the production of solar energy in India", Reading the Wikipedia content on "The Green New Deal". Reflective session on the prospects of "The Green New Deal in India"

Writing Skills: Letter Writing (Formal & Informal) and Hands on Session on Letter Writing

UNIT-III: **(06 Contact Hours)**

Theory: Transport - Problems & solutions

Group Discussion on "The Future of Bullet Trains in India"

PPT on "The Dedicated Freight Corridors & the Future of Indian Economy" – Introduction to Speech

Spoken Skills: Sounds – Vowels, Consonants and Diphthongs – Pronunciation Exercises (Basic Level)

UNIT-IV: **(06 Contact Hours)**

Theory: Technology - Evaluating technology

PPT on "3R: Reduce, Recycle, Reuse" - Solo Debate on "Can Block Chain Technology Mitigate the Issue of Cyber Crimes and Hacking?"

Presentation Skills: JAM –Description of Pictures, Photographs, Process, Talking about wishes, Information Transfer

UNIT-V: **(06 Contact Hours)**

Theory: Environment - Ecology versus Development

Listening Skills: Listening Activity on YouTube video on "Greening the Deserts" - Students' seminar on "Waste to Wealth: Examples from around the Globe".

UNIT-VI: **(06 Contact Hours)**

Theory: Industry - Selling products

Reading Skills: Reading the material on "4Ps: Product, Price, Place, and Promotion" Role play on "How to sell your product and services"

References:

1. Non – Detailed Text Book: Panorama – A Course on Reading published by Oxford University Press, India
2. English for engineers and technologists by Orient Black Swan
3. A Textbook of English Phonetics for Indian Students 2nd Ed T. Balasubramanian. (Macmillan), 2012.
4. Speaking English Effectively, 2nd Edition Krishna Mohan & NP Singh, 2011. (Macmillan).
5. A Hand book for English Laboratories, E.Suresh Kumar, P.Sreehari, Foundation Books,2011
6. English Pronunciation in Use. Intermediate & Advanced, Hancock, M. 2009. CUP
7. Basics of Communication in English, Soundararaj, Francis. 2012.. *New Delhi: Macmillan*
8. English Pronouncing Dictionary, Daniel Jones Current Edition with CD.Cambridge, 17th edition, 2011.

Course outcomes: At the end of the course, the student will be able to

CO 1	Understand the issues affecting the economy and environment in India and across the globe
CO 2	Develop the instinct for problem solution
CO 3	Develop the ability to collect materials on various socio-economic-technological issues and prepare PPT for presentation
CO 4	Improving listening skills
CO 5	Inculcate speaking as a behaviour by repeated practice and exposure

Assessment Method:

Course Nature: THEORY + LABORATORY

Internal Assessment (40 Marks)		**External Assessment (**60 Marks)	
Record Writing	– 10 Marks	Reading Comprehension	– 15 Marks
Attendance	– 10 Marks	Writing	– 30 Marks

Continuous Assessment (Listening – 10 Marks + Oral Presentations – 10 Marks)	Speaking (Viva-Voce) – 15 Marks

****

Course Code	Course Name	Course Category	L-T-P	Credits
20EE1180	Basic Electrical and Electronics Engineering Laboratory	**ESC**	**0-0-3**	**1.5**

Course Learning Objective:

1. To make student get familiarized with the electrical and electronic measuring equipment.
2. To make understand the student the concepts of characteristics of Resistors, Capacitors and Inductors.
3. To understand the behaviour of electrical equipment.
4. To understand the concepts of diodes, transistors and amplification.

List of Experiments:

Familiarization with Computer Hardware and software installation, DSO, Function generators, RPS, FPS, Mustimeters and other lab equipment

Section A: Computer Hardware and software installation:

1. Every student should Draw the block diagram of the CPU along with the configuration of each peripheral and submit to your instructor. Every student should disassemble and assemble the PC back to working condition.
2. Every student should individually install operating system like Linux or MS windows on the personal computer. The system should be configured as dual boot with both windows and Linux.
3. Hardware Troubleshooting: Students have to be given a PC which does not boot due to improper assembly or defective peripherals. They should identify the problem and fix it to get the computer back to working condition.
4. Software Troubleshooting: Students have to be given a malfunctioning CPU due to system software problems. They should identify the problem and fix it to get the computer back to working condition

Section B: Electrical Engineering Laboratory:

1. Verification of ohm's law, series and parallel circuits
2. Verification of Kirchhoff's Laws
3. Verification of Voltage division and Current division principles
4. Verification of circuit theorems
5. V-I characteristics of Incandescent and CFL lamp
6. V-I characteristics of Fluorescent lamp
7. A.C analysis of series R-L circuit and R-C circuit
8. Calibration of Energy meter
9. Open circuit characteristics of D.C Generator
10. Speed control of D.C shunt Motor
11. Three phase power measurement
12. Lab project

Section C: Electronics Engineering Laboratory:

1. Familiarization with any CAD tools like multisim/Pspice/ngspice for doing basic experiments .
2. V-I characteristics of a P-N junction diode and zener diode
3. Half wave and center tapped full wave rectifier
4. Full wave bridge Rectifier with and without filters.
5. Design of a simple amplifier using BJT
6. Experiment on simple analog-modulation scheme
7. Simple experiment on Arduino kit and interfacing with sensors
8. Lab project

Course outcome

At the end of the course, the student will be able to

CO 1	Design basic circuits using P-N junction diode and Zener diode
CO 2	Design rectifier circuits considering the practical aspects into consideration
CO 3	Design simple amplifier with required gain
CO 4	Use circuit knowledge in analyzing Arduino boards
CO 5	Designing simple experiments using Arduino board and sensors interfacing
CO 6	Experimental verification of basic circuit laws and circuit theorems
CO 7	Experimental analysis of V-I characteristics of different electrical and electronic equipments
CO 8	Experimental analysis of electrical machines likes motors, generators etc
CO 9	Design of a simple prototype project

Assessment Method

Assessment Tool	Experiments	Report/Viva-Voce/ Quiz/MCQ/Lab project	Total
Weightage (%)	25%	15%	40%
End Semester Examination weightage (%)			60%

****

Course Code	Course Name	Course Category	L-T-P	Credits
20CS1181	**Problem Solving and Programming Through C Lab**	**ESC**	**0-0-3**	**1.5**

Course Learning Objective:

1. Identify situations where computational methods and computers would be useful.
2. Given a computational problem, identify and abstractthe programming task involved.
3. Approach the programming tasks using techniques learned and write pseudo-code.
4. To understand the concepts of Programming language
5. To learn the basics of C declarations, operators and expressions
6. To learn on the manipulation of strings, functions and pointers
7. To apply concepts and techniques for implementation

List of Programming Assignments for Laboratory:

Statements, Expressions & Conditionals

1. Write a program to print the memory allocation required for all the datatype in C Language.

2. Write a program to check whether the given number is even number or odd number.
3. Write a menu based program to take of input of two values followed input of choice and accordingly perform arithmetic operations like Addition, Subtraction, Multiplication, Modulus, Division, Power(Using Switch Statement)
4. Write a program to swap two given numbers with and without using extra variable.
5. Write a program to find out the whether the given number is a perfect square or not.
6. Write a program to find out whether the given number is positive, negative or zero value.

Iterative Constructs - I: For Loop, While Loop &Do. While

1. Write a program print all the factors of a given number
2. Write a program to find the factorial of a given number
3. Write a program to find whether a given number is Palindrome or not.
4. Write a program to find whether a given number is Prime or not.
5. Write a program to print the Fibonacci series upto given 'n' number of terms.

Iterative Constructs – II: Nested Loops

1. Write a program to print the first 'n' prime numbers and prime numbers upto 'n' value.
2. Write a program to print the Pascal Triangle for given 'n' value
3. Write a program to print the first 'n' perfect number for a given 'n' value.
4. Write a program to print the following pattern for given 'n' value.
5. For Eg. If n = 3, the output would be

```
        *
    *   *   *
*   *   *   *   *
    *   *   *
        *
```

6. Write a program to print the following pattern for given 'n' value
   For Eg. If n = 4, the output would be

```
            2
                3   5
            7   11  13
17  19  23  29
```

Single Dimensional Arrays: Basic Operations and Problems

1. Write a program to take an input array of 'n' numbers and find out the sum of all the elements, product of all the elements and mean of the array.
2. Write a program to take an input array of 'n' numbers and print the second smallest and second largest element of all elements in the array.

Two Dimensional Arrays –Matrices& its operations

1. Write a program to find the addition and subtraction for the given two matrices of sizes 'M x N' and 'P x Q' respectively
2. Write a program to find the multiplication of the given two matrices of sizes 'M x N' and 'P x Q' respectively.
3. Write a program to find transpose of a matrix.

Strings – Dealing with non-numerical data

1. Write a program to convert the Lower Case letters to Upper Case Letters and Upper Case Letters to Lower Case Letters in a given input string.
2. Write a program to the print out the number of vowels, consonants, and digits (0-9) present in the given input string.
3. Write a program to check whether the given input string is palindrome string or not
4. Write a program to sort the given string of characters.

Array of Strings

1. Write a program to find the strings starting with "c" and "a" for the given n input strings..
2. Write a program to print the words of given input string in reverse order For
   Eg. If input string is "I am an Indian", the output would be "Indian an am I"
3. Write a program to arrange the given 'n' strings in Dictionary Order.

Functions

1. Write a program to implement the string operations like Length of String, String Copying, String Concatenation, Conversion to Uppercase and String Comparison.(Define own Function for each of the operation. Header file "string.h" is not allowed)

2. Write a C program to implement Multiplication and Division Operations without using operators "*" and "\" respectively. Define function "mul" for multiplication and "div" for integer division.

Recursion

1. Write a program to print the integers from 1 to N and then N to 1 for the given input number 'N' without using any loops.
2. Write a program to find the X power N(X^N) using the user defined recursive function "pow(X,N)" without using any predefined function from the library.
3. Write a program to find the GCD of two numbers 'a' and 'b' by defining a recursive function GCD(a,b).

Structures

1. Write a program to take the information of 'n' Students (REGID, Name, CGPA, Address – Village, District, Phone NO) and print the topper among the n students.
2. Write a program to take the information of 'n' Students (REGID, Name, CGPA, Address – Village, District, Phone NO) and print the students in the ascending order of Regn ID.
3. Write a program to take the information of 'n' Students (REGID, Name, CGPA, Address – Village, District, Phone NO)and print the list of Phone Number for the students who are the above average of CGPA.

File Handling – Create, Read and Write operations on File

1. Write a program to print the number of lines and words in a given input file name.
2. Write a program to copy from the given file to another file.
3. Write a program to append one file at the end of another file.
4. Write a program to search for a word in a given text file.

Course outcomes

At the end of the course, the student will be able

CO 1	To formulate the algorithms for simple problems
CO 2	To translate the given algorithms to a working and correct program
CO 3	To identify and correct logical errors encountered at run time
CO 4	To write iterative as well as recursive programs
CO 5	To represent Data in arrays, strings, Structures and manipulate them through a program
CO 6	To decompose a problem into functions and synthesize a complete program
CO 7	To be able to create, read and write to and from text files

Assessment Method

Assessment Tool	Experiments	Report/Viva-Voce/ Quiz/MCQ/Lab project	Total
Weightage (%)	25%	15%	40%

End Semester Examination weightage (%)	60%

****

INDIAN CONSTITUTION

Course code	Course name	Course Category	L-T-P	Credits
20HS1101	**Indian Constitution**	**MC**	**2-0-0**	**0**

Course Learning Objectives:

1. The basic objective of the course is to provide knowledge about institutions
2. It help to understands the processes to governing the society in a systematic way.
3. It helps to establish social Justice, Liberty, Equity and Fraternity.
4. The course will introduce the idea of political system in general
5. It provides idea about working process of constitutional institutions.
6. To create awareness about the functioning of the judicial system in India.

Course Contents:

UNIT I: **(5 hours)**

Introduction-Constitution' meaning of the term, Indian constitution sources and constitutional history, Features: Citizenship, Preamble, Fundamental Rights and duties, Directive Principles of State Policy.

UNIT II: **(5 hours)**

Union Government and its Administration-Structure of the Indian Union: Federalism, centre-state relationship, President: Role, power and position, PM and Council of ministers, Cabinet and Central Secretariat, Lok sabha, Rajya sabha.

UNIT III: **(5 hours)**

Election commission- Election commission: Role and functioning, Chief Election Commissioner and Election Commissioners, State Election Commission: Role and functioning, Institute and Bodies for the welfare of SC/ST/OBC and women.

UNIT IV: **(3 hours)**

State Government and its Administration- Governor: Role and position, CM and Council of ministers, state secretariat: Organization, structure and functions.

UNIT V: **(7 hours)**

Local Administration-District's Administration head: Role and importance, Municipalities: Introduction, Mayor and role of Elected Representatives, CEO of Municipal Corporation, Panchayati raj: Introduction, PRI: Zilla Panchayat, Elected officials and their roles, CEO Zilla Panchayat: Position and role, Block level: Organizational Hierarchy (different departments), Village level: Role of elected and appointed officials, Importance of grass root democracy.

UNIT VI: **(5 hours)**

Union Judiciary-Establishment and constitution of Supreme court, Appointment of Judges, Establishment of State High court, Establishment of common High court for 2 or more states, WRITS, PIL(Public Interest Litigation).

Learning resources

Text book:

1. Durga Das Basu, *Constitutions of India*, 23rd ed, LexisNexis Publication.

Reference Books:

1. 'Indian Polity' by Laxmikanth
2.'Indian Administration' by Subhash Kashyap
4.'Indian Administration' by Avasti and Avasti
5.'Government and Politics of India' by W.H.Mrrison Jones
6.'Constitution of India' by J.C.Johari

Course outcomes: At the end of the course, the student will be able to

CO 1	The students will understand their fundamental rules and duties.
CO 2	The students will learn the political system and the system of elections in India.
CO 3	It is to provide the students the institutions and processes to govern themselves in the manner they prefer.
CO 4	Students can also be able to utilize the laws and facilities provided by constution
CO 5	It will provide over all idea about our legal system.
CO 6	It will enable students more strong in terms of law and practice in day to day life.

****

ENGINEERING FIRST YEAR: SEMESTER-2

Course code	Course Name	Course Category	L-T-P	Credits
20MA1202	**Discrete Mathematics**	**BSC**	**3-1-0**	**4**

Course learning objectives: The objective of this course is to

1. Develop mathematical maturity of students to build the ability to understand and create mathematical arguments and to teach them how to think logically and mathematically.
2. Prove theorems and Mathematical arguments by using different methods. provide the mathematical foundations for many computer science courses including data structures, algorithms, database theory, automata theory, formal languages, compiler theory, computer security and operating systems.
3. Learn the basic properties of sets and how to work with discrete structures, which are abstract mathematical structures used to represent discrete objects and relationship between these objects.
4. Introduce basic techniques of counting so that they develop the ability to enumerate..
5. Learn the concepts of graphs and its properties, solving real world problems by using graph concepts.
6. Learn the concepts of Euler Paths , graph coloring , trees.

Course Content:

Unit – I **(10 Contact hours)**

Propositional logic:
Propositions and Connectives, well-formed formulas, Logical Equivalence and laws of logic,Normal forms, PCNF, PDNF.

Unit - II **(10 Contact hours)**

Proof techniques:

Tautological implications and rules of inferences, Methods of proofs(Forward proof, proof by contradiction, contra positive proofs, proof of necessity and sufficiency, Proof by Mathematical induction)

Unit - III **(12 Contact hours)**

Sets, relations and functions:
Sets, Relations, Equivalence Relations and compatibility relations, Transitive closure, Posets, Finite and infinite sets, countable and uncountable sets (definitions), Functions.

Unit - IV **(12 Contact hours)**

Introduction to counting:

Counting Principles, Pigeon hole Principle, Permutations and Combinations, Recurrence Relations, Linear Recurrence relations, Generating functions.

Unit - V **(9 Contact hours)**

Introduction to Graph Theory:
Graphs and their basic properties, Special types of graphs and representations of graphs, Isomorphism's, connectivity.

Unit – VI
Graph Theory(Continuation) : **(7 contact hours)**
Euler and Hamiltonian Paths, Planar Graphs, Graph coloring, Trees

Learning resources:

TEXT BOOKS:

1. Kenneth H. Rosen,*'Discrete Mathematics and its Applications'*, Tata McGraw-Hill. Seventh edition

REFERENCE BOOKS:

1. Trembley and Manohar, '*Discrete Mathematical Structures to Computer Science'*, by Mc - Graw Hill (1997).
2. Kolman, Busby and Ross, *'Discrete Mathematical Structures'* PHI (2009), Sixth Edition.
3. Thomas Koshy, '*Discrete Mathematics with Applications'*, Elsevier Academic press.

Web resources:

1. NPTEL Lectures by Prof. Kamala Krithivasan, Dept of CSE,IIT Madras
   link: https://www.youtube.com/watch?v=xlUFkMKSB3Y&list=PL0862D1A947252D20
2. MIT open course ware: Mathematics for Computer Science, Fall 2010. Instructor: Tom Leighton

link: https://www.youtube.com/watch?v=L3LMbpZIKhQ&list=PLB7540DEDD482705B

Also visit: http://ocw.mit.edu/6-042JF10

3. Discrete Mathematics for GATE. IIT lecture:

link:
https://www.youtube.com/watch?v=E6uhC0pT9J8&list=PLEJxKK7AcSEGD7ty8DB1aU0xVG_P_hs_0
4.RGUKT Course Content

Course outcomes: At the end of the course, the student will be able to

CO 1	Read, comprehend and construct mathematical argument
CO 2	Prove theorems and mathematical statements in different techniques.
CO 3	Deal with set, relation, countability and functions.
CO 4	Apply permutation, combination, pigeon hole principle, recurrence relation and generating functions to enumerate objects.
CO 5	Understand and apply concepts of graph in many computer science courses.
CO 6	Deal with Euler paths in graphs and coloring of graphs

Assessment method for Theory courses only:

Course Nature	**Theory**			
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

****

Course code	**Course name**	**Course Category**	**L-T-P**	**Credits**
20PY1201	**E1 Engineering Physics-CSE**	**BSC**	**3-1-0**	**4**

Course Learning Objectives:

1. To enhance the knowledge on waves & oscillation with an emphasis on different type of oscillations and its resonance conditions.
2. To distinguish vividly the Optical phenomena's such as Interference, Diffraction and their applications through experimental point of view like Michelson Interferometer, Newton Rings and Plane Diffraction Grating.
3. To pursue the in-depth knowledge on Polarization with emphasis on Laurent's half-shade Polarimeter and also to learn all the basic necessary concepts regarding the LASERs including basic important types of LASERs.
4. To pull the student attention towards the difference between Photography and Holography and also the basic knowledge regarding Optical fibers along with its applications.
5. To identify the necessity of origin of Quantum mechanics over the grand old Classical mechanics and also to learn the knowledge on postulates of Quantum mechanics.

6. To get knowledge about the band theory of solids by the assumption of movement of an electron in the periodic potential well only and hence distinguishes the materials classification, phenomena of Hall Effect exhibited by semiconductors and its applications.

Course Content:

Unit I: Oscillations (8 Hours)

Oscillations**:** Simple Harmonic Oscillator (SHO), Damped Oscillations, Forced Oscillations, Amplitude and Velocity Resonance, Quality Factor, Coupled Oscillations &Normal modes, Coupled Pendulums & energy and Oscillation on N coupled modes.

Unit – II: Wave Optics (10Hours)

Interference**:** Superposition principle, Division of amplitude and wave front division, Interferometers (Michelson), Newton's Rings due to Reflected waves, Applications; Diffraction: Fraunhoffer diffraction (single, double & multiple slits),Plane Diffraction Grating, Rayleigh criterion for resolving power, Dispersive power, Applications.

Unit – III: Polarization and LASERS (9 Hours)

Polarization: Classification of Polarized light: Linear, Circular, Elliptical; Production & detection of polarized light; Retardation wave plates: Quarter & Half wave plates; Optical activity: Laurent Half shade Polarimeter; Basic principles of Lasers, Theory of Lasers, and Types of Lasers: Three level and four level lasers, Ruby Laser, He-Ne Laser, and Semiconductor laser: P-N Junction Diode Laser, applications of lasers.

Unit – IV: Holography, and Optical fibers (9 Hours)

Basic principles of Holography, types of holograms, difference between photography and holography, holographic NDT &applications of holography; Optical fibers: Basic principles, types and applications for communication and sensing, Acceptance angle &Numerical Aperture NA.

Unit V: Quantum Mechanics (12 Hours)

Photo electric effect, Compton effect; De-Broglie matter waves, properties of matter waves, Uncertainty Principle, Wave function& it's probability interpretation, Operators, Expectation values, Postulates of quantum mechanics, Time independent Schrodinger Equation and its Applications, Particle in a box (1-D and 3-D).

Unit VI: Semiconductor Physics (12 Hours)

Electron in periodic structures, Band theory of solids, E-K curve, effective mass, Density of states, Fermi levels. Intrinsic and extrinsic semiconductors, dependence of Fermi level on carrier-concentration and temperature (equilibrium carrier statistics), Carrier generation and recombination, Carrier transport: diffusion and drift, Hall effect- Hall sensors, Physics of p-n junction, Metal-semiconductor junction (Ohmic and Schottky)

Text Books:

1. Dr. N. Subrahmanyam, Brijalal, Dr. M.N Avadhanulu"A Text Book of Optics" S. chand Publication
2. Md. N. Khan & S. Panigrahi "Principals of Engineering Physics" Volume I, Volume II, Cambridge University Press
3. Hitendra K. Malik and A.K. Singh, '*Engineering Physics*'Tata McGraw Hill, 2nd Edition, 2017
4. Gaur and Gupta "*Engineering Physics*, Dhanpathrai Publications, 6th edition

References:

1. AjoyGhatak'*Optics*' Tata McGraw Hill, 6th Edition
2. M. Armugam, Anuradha'*Engineering Physics*', Agencies publishers, 2003

3. David McMahon, '*Quantum Computing Explained'*, Wiley, 2016

Course outcomes: At the end of the course, the student will be able to

1: Student will be able to distinguish the phenomena's of interference & diffraction exhibited by light waves theoretically through Michelson Interferometer, Newton's Rings and Plane Diffraction Grating.

2: Student will have capable to understand the lengths and breadths of Concept called Polarization as well as working nature and construction of LASERs rather closely along with its applications in various fields.

3: Students will have capable to discriminate the merits and de-merits of Holography over the Photography besides of that they can purse the knowledge about optical fibers and their applications.

4: Student will be able to differentiate all type of oscillations like Simple Harmonic, Forced, Damped & Coupled and also implications governed by Amplitude & Velocity Resonance.

5: Student will be able to construct a quantum mechanical model to explain the behavior of a system at microscopic level and solve engineering problems using the laws of quantum mechanics.

6: Student will acquire the capacity to describe classification of solid state materials by the band theory of solids and semiconducting materials along with its significance.

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

****

MANAGERIAL ECONOMICS AND FINANCIAL ANALYSIS

Course code	**Course name**	**Course Category**	**L-T-P**	**Credits**
20BM1201	**Managerial Economics and Financial Analysis**	**HSC**	**3-0-0**	**3**

Course Learning Objectives:
1. To strengthen students managerial skill.
2. To enhance the conceptual clarity in economic concepts.
3. To develop to forecasting capability.
4. It will help to produce multi-disciplinary thought.
5. It will enhance their conceptual and practical/hand on practice in accounting.
6. It will help to implement and understand the uses of ratios.

Course Contents:

Unit I: **(7 hours)**
Introduction to managerial economics, consumer behavior, demand, demand analysis, demand forecasting, supply, supply analysis.

Unit II: **(7 hours)**
Theory of production, production functions, concept of cost, cost analysis, break even analysis.

Unit III: **(7 hours)**
Market structure-monopoly, oligopoly, monopolistic, prefect market; Types of business organizations-sole proprietorship, partnership, private ltd. Companies and public ltd. Companies, formation of company.

Unit IV: **(8 hours)**
Introduction to capital, capital sources, capital budgeting- NPV, IRR, Payback period, profitability index.

Unit V: **(8 hours)**
Introduction to financial accounting, rules of debit-credit, Double-Entry Book Keeping, Journal, Ledger, Trial Balance- Final Accounts (Trading Account, Profit and Loss Account and Balance Sheet with simple adjustments, Preparation of final account and other related accounting statements.

Unit VI: **(8 hours)**
Financial statements, comparative statement analysis, common- size statement analysis, ,ratio analysis, time series (only theories).

Learning resources
Text book:
1. 1. Aryasri, A. R., *Managerial Economics & Financial Analysis*, McGraw Hill, 2014.

Reference Books:
1. Siddiqui., *Managerial Economics & Financial Analysis*, 2e, New Age International Private Limited, 2017.
2. . Pandey, I.M., "*Financial Management*", 11e, Vikas Publishing House, 2015.
3. . Prasanna Chandra., "*Financial Management: Theory and Practice*", 9e, Mc Graw Hill Education, 2015.
Web resources:

1. Managerial Economics and Financial Analysis, Dr. Trupti , IIT Bombay
http://nptel.ac.in/courses/110101005/

Course outcomes: At the end of the course, the student will be able to

CO 1	A student will be able to understand basic economics as well as management concepts.
CO 2	This subject will provide implication facilities of concepts.
CO 3	Students can be able to do primary data collection and classification.
CO 4	Students can also be able to forecast as well as generate trend series by utilizing the available secondary data.

CO 5	They have basic knowledge about accounting and its terminologies.
CO 6	They will be able to prepare and understand accounting tables.

For Theory courses only:

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

****

Course code	**Course Name**	**Course Category**	**L-T-P**	**Credits**
20CS1201	**Object Oriented Programming through JAVA**	**PCC**	**3-1-0**	**4**

Course Learning Objectives:

1. Gain knowledge about basic Java language syntax and semantics to write Java programs and use concepts such as variables, conditional and iterative execution methods etc.,
2. Understanding the fundamentals of object-oriented programming in Java, including defining classes, objects, invoking methods etc. and exception handling mechanisms.
3. Understand the principles of inheritance, packages and interfaces.
4. Understand the principles of Multithreading and Event handling mechanisms.

Course Content:

Unit 1: (7.5 Contact hours)

Introduction: Object Oriented Programming, Introduction to java and JVM, Key features, Fundamentals of Objects and Classes, Access Specifiers, data types, dynamic initialization, scope and life time, operators, Conditional Statements, control structures, arrays, type conversion and casting. Constructors, usage of static, access control, this key word, garbage collection, overloading, parameter passing mechanisms, nested classes and inner classes.

Unit II: (7.5 Contact hours)

Strings: Exploring the String class, String buffer class, Command-line arguments. Library: StringTokenizer, Random class, Wrapper classes. Encapsulation: Abstraction. Creating User defined Data Structures: Array of Objects, User defined Linked List.

Unit III: (10 Contact hours)

Inheritance and Interface:Types of Inheritance, usage of super key word, method overriding, final methods and classes, abstract classes, Polymorphism: dynamic method dispatch, Static method dispatch. **Interfaces**: Differences between classes and interfaces, defining an interface, implementing interface, variables in interface and extending interfaces.

Unit IV: (6 Contact hours)

File Handling: Streams, File class, File streams. File Reader, File Writer, Buffered Reader, Buffered Writer, String Tokenizer **Exception Handling:** Concepts of Exception handling, types of exceptions, usage of try, catch, throw, throws and finally keywords, Built-in exceptions, creating own exception sub classes.

Unit V (6 Contact hours)

Packages: Creating a Package, setting CLASSPATH, Access control protection, importing packages.**Multithreading :** Concepts of Multithreading, differences between process and thread, thread life cycle, Thread class, Runnable interface, creating multiple threads, Synchronization, thread priorities, inter thread communication, daemon threads, deadlocks, thread groups.

Unit VI (8Contact hours)

Event Handling: Introduction to Event Handling, AWT Components, windows, Layout Managers, Event handling model of AWT, Adapter classes, Menu, Menu bar. **Swings**: swings introduction, JFrame, JPanel and JComponent, Icons and Labels, text fields, buttons – The JButton class, Check boxes, Radio buttons. Combo boxes, Action Listeners. Introduction to JDBC.

Learning Resources

Text books:

- Herbert Schildt, "The Complete Reference Java", TMH Publishing Company Ltd, 9th Edition.
- Cay Horstmann, "Big Java", John Wiley and Sons, 2nd Edition

Reference Books:

- Allen B.Downey,"Think Java; How to Think Like a Computer Scientist",Paper Back 1stEdition
- David J. Eck, Hobart and William Smith Colleges, "Introduction to Programming Using Java" Published by Paper Back .

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

Web resources:

1. http://www.nptelvideos.com/java/java_video_lectures_tutorials.php
2. https://www.tutorialspoint.com/java/
3. https://www.javatpoint.com/java-tutorial
4. http://mooc.fi/courses/2013/programming-part-1/material.html
5. http://math.hws.edu/javanotes

Course Outcomes: At the end of the course, the student will be able to

CO 1	Explain OOP Principles and Write Basic Java Programs.
CO 2	Defining Classes and Objects. Identify classes, objects, members of a class and relationships among them needed for a specific problem
CO 3	To be able to write Java Programs to demonstrate method overloading and Demonstrate the concepts of polymorphism and inheritance. Discuss method overriding V/s method overloading.
CO 4	Explain the benefits of JAVA's Exceptional handling mechanism compared to other Programming Language
CO 5	To be able to write Java Programs to demonstrate Packages and Threading concepts.
CO 6	Discuss and Demonstrate the AWT Concepts and develop the AWT Applications.

For Theory courses only:

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

****

Course Code	Course Name	Course Category	L-T-P	Credits
20CS1202	**Data Structures**	**PCC**	**3-0-0**	**3**

Course Learning Objectives:

1. To understand the basic concepts such as Abstract Data Types, Linear, and Non Linear Data structures
2. To understand the behavior of data structures such as stacks, queues
3. To understand building of trees and its operations
4. To be familiar with searching and sorting algorithms
5. To choose the appropriate data structure for a specified application.
6. To study various graph processing algorithms and Algorithm Design technique

Course Content:

Unit- I **(7 Contact Hrs)**

Introduction to Linear and Non-Linear data structures. Singly Linked Lists-Operations-Insertion, Deletion, Searching, Concatenation of singly linked lists, Circularly linked lists-Operations for Circularly linked lists, Doubly Linked Lists- Operations Insertion, Deletion, Searching.

Unit- II **(8 Contact hours)**

Stack ADT, definition, operations, array and linked implementations in C, applications-infix to postfix conversion, Postfix expression evaluation, recursion implementation, Queue ADT, definition and operations, array and linked Implementations in C, Circular Queues-Insertion and deletion operations, Deque (Double ended queue) ADT, array and linked implementations in C.

Unit- III **(9 Contact hours)**

Sorting-Insertion Sort, Selection Sort, Merge Sort, Quick sort, Heap Sort, Comparison of Sorting methods and linear sorting algorithms-Counting sort, Radix sort, shell sort

Searching – Linear Search, Binary Search, Basic Concepts- Hashing Methods- Collision Resolutions- Open Addressing- Linked List Collision Resolution- Bucket Hashing

Unit- IV **(6 Hrs)**

Trees – Terminology, Representation of Trees, Binary tree ADT, Properties of Binary Trees, Binary Tree Representations-array and linked representations, Binary Tree traversals, Max Priority Queue ADT-implementation-Max Heap-Definition, Insertion into a Max Heap, Deletion from a Max Heap, Min-Heap-Operations on Min-Heap.

Unit- V **(9 Contact hours)** Search Trees-Binary Search Trees, Definition, Operations- Searching, Insertion and Deletion, AVL Trees-Definition and operations on AVL Tree, Red Black Trees, Trie Tree, B and B+ -Trees.

Unit- VI **(6 Contact hours)**

Graphs – Introduction, Definition, Terminology, Graph ADT, Graph Representations- Adjacency matrix, Adjacency lists, Graph traversals – DFS and BFS. Applications of DFS and BFS-Connected Components, Topological sort.

Text Books:

1. R. Thareja “Data Structures using C” , Oxford University Press.
2. M. A. Weiss “Data structures and Algorithm Analysis in C “ , 2nd edition, Pearson.

References

1. Narsimha Karumanchi “Data Structures and Algorithms made easy in C”, 2nd Edition, Career Monk Publications.
2. Allen Weiss, “Data Structures and Algorithms Analysis in C”, Pearson, 2nd Edition

Web resources:

1. Indian Institute of Technology, Madras , “Programming and Data Structures”, URL: https://nptel.ac.in/courses/106106130/
2. Indian Institute of Technology, Delhi, ‘Data Structures and Algorithms’, https://nptel.ac.in/courses/106102064/

Course outcomes: At the end of the course, the student will be able to

CO 1	Learn how to use data structure concepts for realistic problems.
CO 2	Able to use linear and non-linear data structures like stacks, queues , linked list etc.
CO 3	Ability to identify appropriate data structure for solving computing problems in respective language.
CO 4	Ability to implement algorithms for the creation, insertion, deletion, searching, and sorting of each data structure.
CO 5	To build the different Hight balanced trees and perform the basic operations

For Theory courses only:

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

****

Course code	**Course name**	**Course Category**	**L-T-P**	**Credits**
20PY1281	**Engineering Physics Laboratory**	**BSC**	**0-0-3**	**1.5**

Course Learning Objectives:

1. To determine the wavelength of laser light using Diffraction Grating.
2. To determine the radius of curvature of a Plano convex lens by Newton’s Ring experiment.
3. To determine the specific rotation of cane sugar solution with the help of Polarimeter.
4. To calculate the velocity of ultrasonic sound waves in different liquid media by interferometer.
5. To study the phenomena of Hall Effect in given semiconductors and to calculate:-(i) The Hall Coefficient (R_H) (ii) the concentration of charge carriers of given semiconductor material.
6. To verify the postulates of Bohr’s theory and discrete (quantized) energy level of atoms.
7. To study the photoelectric effect and determine the value of Plank’s constant.

8. To determine the Energy Band Gap of a Semiconductor by using a Junction Diode / Four Probe method
9. Study of I-V Characteristic of a solar cell illuminated by an incandescent lamp at different fixed frequencies.
10. Determination of Acceptance angle and Numerical Aperture using fiber optic cable

Experiments list

1. Laser Diffraction
2. Newton's Ring expt
3. Polarimeter.
4. Ultrasonic interferometer
5. Hall Effect
6. Frank Hertz
7. Photo electric Effect
8. Energy Band Gap of a Semiconductor
9. Solar cells
10. Optical fiber

Course outcomes: At the end of the course, the student will be able to

1. Student will able to recognize the diffraction phenomena exhibited by different grating elements and then capable to calculate wavelength of given laser light.
2. Student will able to recognize the interference phenomena exhibited by division of amplitude using Newton's Ring experiment. And capable of calculate the radius of curvature by given Plano-convex lens by ring pattern.
3. Student will able to distinguish the purity of sugar cane by measuring polarization and specific rotation of given sugar solution with the help of Polarimeter. The higher the polarization purer will more in sugar; and lesser the polarization impurity will be more.
4. Student will able to understand the generation of ultrasonic sound waves by piezoelectric effect in different liquid media and to know how the velocity of sound wave varies with adiabatic compressibility of medium.
5. Student will have capable to distinguish the nature of semiconductors by measuring Hall coefficient, carrier density and carrier mobility of a given semiconductor.
6. Students will able to understand quantization of energy in atoms.
7. Student will able to understand physical characteristics of photoelectric effect and how to calculation of Plank's constant value by Einstein particle radiation concept.
8. Student will have ability to describe the relation between conductivity and temperature in semiconductor materials and then calculate the energy gap ofgiven material.
9. Student will have skillful to draw I-V characteristic of solar cells.
10. Student will able to calculate the acceptance angle and numerical aperture using fiber optic cable.
11. Student will able to understand how recording and reconstruction of holograms under Laser light.

Course Nature	**Practical**
Assessment Method	

Assessment Tool	Experiments	Record	Viva-Voce/ Quiz/MCQ/Lab project	Total
Weightage (%)	25%	5%	10%	40%
End Semester Examination weightage (%)				60%

****

Course code	Course Name	Course Category	L-T-P	Credits
20CS2181	**Object Oriented Programming Through JAVA Lab**	**ESC**	**0-0-3**	**1.5**

Course Learning Objective:

1. To build software development skills using java programming for real-world applications.
2. To understand and apply the concepts of classes, packages, interfaces, arraylist, User defined Linked List, File Handling, exception handling and Multi-threading.
3. To develop applications using AWT programming and event handling.

List of Experiments:

1. Lab No 1: Basic Programs in JAVA
2. Lab No 2: Programming Assignments on Arrays and Strings
3. Lab No 3: Programming Assignments on Classes, Objects and Encapsulation
4. Lab No 4: Implementing the concepts of Inheritance and Array Objects
5. Lab No 5: Implementing the OOPS Concepts of Abstract, Interfaces and Polymorphism
6. Lab No. 6: Programming Assignments on File Handling
7. Lab No. 7: Programming Exercises on Exception Handling
8. Lab No 8: Working with List Operations
9. Lab No 9: Implementing the concepts of Multi-Threading
10. Lab No 10: Programming Exercises on Event Handling

Course Outcomes

At the end of the course, the student will be able to

CO 1	Understanding the control structures and conditional statements in Java
CO 2	Understanding the arrays and String handling in java
CO 3	Understanding the difference between class and object and providing security for objects
CO 4	Understanding the reusability of objects and working with multiple objects
CO 5	Understanding about hiding the data, getting multiple inheritance through interfaces
CO 6	Understanding the data processing from files
CO 7	Understanding about handling run time abnormal program executions
CO 8	Understanding about creating user defined linked list and dynamic objects
CO 9	Understanding the multi-threaded programming and inter thread communication
CO 10	Understanding about GUI creation

Assessment Method

Assessment Tool	Experiments	Report/Viva-Voce/ Quiz/MCQ/Lab project	Total

Weightage (%)	25%	15%	40%
End Semester Examination weightage (%)			60%

Course Nature		**Practical**		
Assessment Method				
Assessment Tool	Experiments	Record	Viva-Voce/ Quiz/MCQ/Lab project	Total
Weightage (%)	25%	5%	10%	40%
End Semester Examination weightage (%)				60%

****

Course Code	Course Name	Course Category	L-T-P	Credits
20CS1282	**Data Structures Lab**	**PCC**	**0-0-3**	**1.5**

Course Objectives:

1. To develop skills to design and analyze simple linear and non-linear data structures
2. To strengthen the ability to identify and apply the suitable data structures for the given real-world problem
3. To gain knowledge in practical applications of data structures.

List of Experiments:

1. Write a C program that uses functions to perform the following:
   a) Create a singly linked list of integers.
   b) Delete a given integer from the above linked list.
   c) Display the contents of the above list after deletion.
2. Write a C program that uses functions to perform the following:
   a) Create a doubly linked list of integers.
   b) Delete a given integer from the above doubly linked list.
   c) Display the contents of the above list after deletion.
3. Write a C program implement the Stack ADT using Arrays and Linked List.
4. Write a C program that uses stack operations to convert a given infix expression into its postfix equivalent.
5. Write a C program that evaluates a postfix expression.
6. Write C program to implement queue ADT using array and doubly linked list.
7. a) Write C program to implement priority queue ADT using array.
   b) Write C program to implement circular queue ADT using array.
8. Write C program for implementing the following sorting methods:
   b) Insertion sort b) Merge sort
9. Write C program for implementing the following sorting methods:
   b) Quick sort b) Selection sort
10. Write a C program for implementing Heap sort algorithm.
11. Write a C program that uses functions to perform the following:
    a) Create a Binary Search Tree (BST).
    b) Insert data in BST

b) Traverse the above BST recursively in Postorder.

12. Write a C program that uses functions to perform the following:
    a) Deletion an element BST
    b) Traverse the above BST non recursively in Inorder.

13. Write a C program to implement all the functions of a dictionary (ADT) using hashing.
14. Write C program for implementing Depth first traversal and Breadth first traversal.

Course Outcomes:

At the end of this lab session, the student will

CO 1	Be able to design and analyze the time and space efficiency of the data structure
CO 2	Be capable to identity the appropriate data structure for given problem
CO 3	Have practical knowledge on the application of data structures

Assessment Method

Assessment Tool	Experiments	Report/Viva-Voce/ Quiz/MCQ/Lab project	Total
Weightage (%)	25%	15%	40%

Course Nature		**Practical**		
Assessment Method				
Assessment Tool	Experiments	Record	Viva-Voce/ Quiz/MCQ/Lab project	Total
Weightage (%)	25%	5%	10%	40%
End Semester Examination weightage (%)				60%

****

Course code	Course name	Course Category	L-T-P	Credits
20BE1201	Environmental Science	Mandatory	2-0-0	0

Course Learning Objectives:

1. To provide knowledge about multidisciplinary nature of environment, various sources of natural energy.
2. Understanding of ecosystem structure and function etc.
3. Knowledge of biodiversity and conservation
4. Understanding of problems caused by pollution and its impact
5. Understanding about the various social issues related to environment.
6. Awareness for the Environment and human health

Course Content:

UNIT-I: The Multidisciplinary Nature of Environmental Studies and Natural Resources (9 hours)

The Multidisciplinary Nature of Environmental Studies: Definition, scope and importance; Need for public awareness.

Natural Resources: Renewable and Non Renewable Resources

Natural resources and associated problems.

a) Forest resources: Use and over-exploitation, deforestation, case studies. Timber extraction, mining, dams and their effects on forest and tribal people. b) Water resources: Use and over-

utilization of surface and ground water, floods, drought, conflicts over water, dams-benefits and problems. c) Mineral resources: Use and exploitation, environmental effects of extracting and using mineral resources, case studies. d) Food resources: World food problems, changes caused by agriculture and overgrazing, effects of modern agriculture, fertilizer-pesticide problems, water logging, salinity, case studies. e) Energy resources: Growing energy needs, renewable and non renewable energy sources, use of alternate energy sources. Case studies. f) Land resources: Land as a resource, land degradation, man induced landslides, soil erosion and desertification. Role of an individual in conservation of natural resources. Equitable use of resoureces for sustainable lifestyles.

UNIT-II: Ecosystems (4 hours)

Concept of an ecosystem, Structure and function of an ecosystem, Producers, consumers and decomposers,Energy flow in the ecosystem,Ecological succession, Food chains, food webs and ecological pyramids,Introduction, types, characteristic features, structure and function of the following ecosystem:-a. Forest ecosystem, b. Grassland ecosystem, c. Desert ecosystem, d. Aquatic ecosystems (ponds, streams, lakes, rivers, oceans, estuaries).

UNIT-III: Biodiversity and It's Conservation (4 hours)

Introduction – Definition: genetic, species and ecosystem diversity, Biogeographical classification of India,Value of biodiversity: consumptive use, productive use, social, ethical, aesthetic and option values, Biodiversity at global, National and local levels, Inida as a mega-diversity nation, Hot-sports of biodiversity, Threats to biodiversity: habitat loss, poaching of wildlife, man-wildlife conflicts, Endangered and endemic species of India, Conservation of biodiversity: In-situ and Ex-situ conservation of biodiversity.

UNIT-IV: Environmental Pollution (6 hours)

Cause, effects and control measures of:-a. Air pollution, b. Water pollution, c. Soil pollution, d. Marine pollution, e. Noise pollution, f. Thermal pollution, g. Nuclear hazards, Solid waste Management: Causes, effects and control measures of urban and industrial wastes, Role of an individual in prevention of pollution, Pollution case studies, Disaster management: floods, earthquake, cyclone and landslides.

UNIT- V: Social Issues and the Environment (4 hours)

From Unsustainable to Sustainable development Urban problems related to energy, Water conservation, rain water harvesting, watershed management, Resettlement and rahabilitation of people; its problems and concerns. CaseStudies, Environmental ethics: Issues and possible solutions.• Climate change, global warming, acid rain, ozone layer depletion, nuclear accidents and holocaust. Case Studies. Wasteland reclamation, Consumerism and waste products, Environment Protection Act, Air (Prevention and Control of Pollution) Act, Water (Prevention and control of Pollution) Act, Wildlife Protection Act, Forest Conservation Act, Issues involved in enforcement of environmental legislation, Public awareness.

UNIT-VI: Human Population and the Environment (3 hours)

Population growth, variation among nations, Population explosion – Family Welfare Programme, Environment and human health, Human Rights, Value Education, HIV/AIDS, Women and Child Welfare, Role of Information Technology in Environment and human health, Case Studies.

Learning Resources

Text Book:

1. Erach Bharucha, '*Textbook of Environmental studies',* UGC

Reference Books:

1. Clark RS, '*Marine Pollution'*, Clanderson Press, Oxofrd (TB).
2. De AK, '*Environmental Chemistry'*, Wiley Eastern Ltd.

Course Outcomes: At the end of the course, the student will be able to

CO1	Well understanding about their surrounding natural resources and their conservation
CO 2	Able to understand the ecosystem food chain and habitat.
CO 3	Develop the practices for conservation of biodiversity
CO 4	To well understand the pollution courses, impact and prevention from pollution
CO 5	Able to bring about an awareness of a variety of environmental concerns.
CO 6	It attempts to create a pro-environmental attitude and a behavioral pattern in society that is based on creating sustainable lifestyles.

For Theory Courses Only:

Course Nature	**Theory**			
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

****

ENGINEERING SECOND YEAR:SEMESTER-1

Course code	**Course Name**	**Course Category**	**L-T-P**	**Credits**
20MA2102	**Probability and Statistics (CSE)**	**BSC**	**3-1-0**	**4**

Course Learning Objectives:
1. Providing students with a formal treatment of probability theory.
2. Equipping students with essential tools for statistical analysis.
3. Fostering understanding through real-world statistical applications.
4. Develop skills in presenting quantitative data using appropriate diagrams, tabulations.
5. Use appropriate statistical methods in the analysis of simple datasets.
6. Instill the belief that Statistics is important for scientific research.

Course Content:

Unit – I
Probability and theorems in Probability (8 Contact hours)
Probability introduction through Sets and Relative Frequency, Experiments and Sample Spaces, Discrete and Continuous Sample Spaces, Events, Probability Definitions and Axioms, Mathematical Model of Experiments, Probability as a Relative Frequency, Joint Probability, Conditional Probability, Total Probability, Bayes' Theorem and Independent Events.

Unit – II
Probability Distributions: (10 Contact hours)

Discrete distributions: Bernoulli, Binomial, Poisson, Negative Binomial, Geometric and hyper geometric distributions (Find their mean, variance and problems). Continuous distributions: Uniform, Exponential, Normal, Beta and Gamma distributions.

Unit – III

Moment Generating functions **(10 Contact hours)**

Functions of Random Variables, Correlation coefficient and Bivariate Normal Distribution. Probability Inequalities and Generating Functions, Moment Generating Function, Characteristic Function, Cumulant Generating Function, Probability Generating Function.

Unit – IV

Order Statistics and Central Limit theorem **(08 Contact hours)**

Order Statistics, Convergence of Sequence of Random Variables, Weak Law of Large Numbers, Strong Law of Large Numbers, Central Limit Theorem.

Unit - V

Sampling Theory: **(12 Contact hours)**

Definition of population, sampling, statistics and parameters. Types of sampling, Expected values of sample mean and variance, sampling distribution, standard error, sampling distribution of mean and sampling distribution of variance. Sampling -Distributions (t, F and Chi-square), confidence interval and interval estimation.

Unit – VI

Large Sample Tests: **(12 Contact hours)**

Definition of Null and alternative hypothesis, critical region. Type I and Type II errors, power of the test, one tail, two tail tests, Tests for the single mean, two means, single proportion and two proportions using Z-test and t-test, t-test and F-test for significance of difference variance.

Learning resources

Text book:

1 William W. Hines and Douglas C. Montgomery, '*Probability and Statistics in Engineering'*, Willy Publications, 4th Edition.

Reference Books:

1. Sheldon Ross, '*A First Course in Probability'*, Pearson Publications, 9th Edition.
2. Athanasios Papoulis and S. Unnikrishna Pillai, '*Probability, Random Variables and Stochastic Processes'*, TMH, 4th Edition,.

Web resources:

1. https://nptel.ac.in/courses/117105085/
2. https://nptel.ac.in/courses/111106112/
3. https://nptel.ac.in/courses/111102111/
4. RGUKT Course Content

Course outcomes: At the end of the course, the student will be able to

CO 1	Apply Probability theory via Bayes Rule.
CO 2	Describe the properties of Discrete and Continuous distributions.
CO 3	Apply problem-solving techniques to solving real-world events.
CO 4	Apply selected probability distributions to solve problems.
CO 5	Develop problem-solving techniques needed to accurately calculate probabilities.

CO 6	Interpret and clearly present output from statistical analysis.

Probability and Statistics	**Theory**			
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weight age (%)	10%	30%	60%	100%

****

Course code	Course Name	Course Category	L-T-P	Credits
20CS2103	**Formal Languages and Automata Theory**	**PCC**	**3-0-0**	**3**

Course Learning Objectives:

1. **To understand and design Finite State Machines and applications.**
2. **To Understand about Regular Expressions and its applications.**
3. **Understanding of formal grammars and their applications.**
4. **Understanding various other formal languages and their designing models.**
5. **To understand Decidability and Undecidability of various problems in the theoretical computer science.**

UNIT I: Introduction to Automata (6 Contact hours)

Strings, Alphabet, Language, Operations, Finite state machine, definitions, finite automaton model, acceptance of strings, and languages, deterministic finite automaton and non-deterministic finite automaton, transition diagrams and Language recognizers.

UNIT II: Finite Automata (9 Contact hours)

NFA with Ɛ-transitions - Significance, acceptance of languages. Conversions and Equivalence: Equivalence between NFA with and without Ɛ transitions, NFA to DFA conversion, minimization of FSM, equivalence between two FSM's, Finite Automata with output- Moore and Mealy machines, Equivalence between Moore and Mealy.

UNIT III: Regular Languages (6 Contact hours)

Regular sets, regular expressions, identity rules, Constructing finite Automata for a given regular expressions, Conversion of Finite Automata to Regular expressions. Pumping lemma of regular sets, Closure properties of regular sets (Proofs not required).

UNIT IV: Grammars (9 Contact hours)

Regular grammars: Right linear and left linear grammars, Equivalence between regular linear grammar and FA, Inter conversion, Context free grammar, derivation trees, and sentential forms. Right most and leftmost derivation of strings.

Context Free Grammars: Ambiguity in context free grammars. Minimization of Context Free Grammars. Chomsky Normal Form, Greibach Normal Form, Pumping Lemma for Context Free Languages. Enumeration of properties of CFL (Proofs omitted).

UNIT V: Push Down Automata (7 Contact hours)

Push down automata, definition, model, acceptance of CFL, Acceptance by final state and acceptance by empty state and its equivalence. Equivalence of CFL and PDA, Inter conversion. (Proofs not required). Introduction to DCFL and DPDA.

UNIT VI: Turing Machine & Computability Theory (8 Hrs)
Recursive and Recursively enumerable languages, and Church's Hypothesis. Turing Machine: Introduction, Components of Turing Machine, Description of Turing Machine, Elements of TM, Language accepted by a TM, Role of TMs, Design of TMs, Universal Turing Machine, Undecidability of Post Correspondence problem.

Text Books:
Hopcroft, J D Ullman "Introduction to Automata and Language Theory", 3rd Edition, 2006
C. Papadimitrou and C. L. Lewis. Elements of Theory of Computation, Prentice-Hall, 1981.

Reference Books:
John.C.Martin, "Introduction to Languages and the Theory of Computation" McGraw-Hill Education, 01- May-2010.
Kamala Krithivasan, Rama.R, "Introduction to Formal Languages, Automata Theory and Computation", Pearson Education India, 01-Sep-2009

Web Resources
Indian Institute of Technology, Guwahati, " Formal Languages and Automata Theory", https://nptel.ac.in/courses/111103016/

Course outcomes: At the end of the course, the student will be able to

CO 1	Construct finite state diagrams while solving problems of computer science
CO 2	Ability to convert NFA to DFA and Epsilon NFA to DFA
CO 3	Ability to convert RE to Finite Automata and vise versa
CO 4	Design of new grammar and language
CO 5	Ability to design PDA and NPDA
CO 6	Ability to learn design of Turing machine and Find solutions to the problems using Turing machines

For Theory courses only:

Course Nature	**Theory**			
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

Course Code	**Course name**	**Course Category**	**L-T-P**	**Credits**
20EC2110	**Digital Logic Design**	**ESC**	**3-0-0**	**3**

Course Learning Objectives:
1. To discuss the relevance of Digital Logic Design with Computer Science and Engineering course.

2. To discuss the concepts of Number systems and representations, combinational design, sequential designs and complete system design at gate-level abstraction in computer Design.
3. To discuss the important features of IC design like area, power and delay.

Course Content

Unit-I **(8Contact hours)**

Number systems-Representations-Conversions, error detection and error correction, Boolean constants and variables, basic gates: operation and truth tables, describing logic gates algebraically, evaluating logic circuit outputs, implementing circuits from Boolean expressions, universality of gates, Boolean theorems

Unit-II **(8 Contact hours)**

Combinational circuit minimization using Boolean laws and karnaugh maps, multilevel synthesis, logic levels and noise margins. Single bit adders and subtractors, parallel adders, multi-bit subtraction using adders, signed multiplier, unsigned multiplier

Unit-III **(6 Contact hours)**

Decoders, Encoders, Multiplexers, Demultiplexers. Realization of various functions using Decoders, Multiplexers. Priority encoders ,IC 74x148

Unit-IV **(7 Contact hours)**

Bistable elements, Latches and Flip-flops : S-R latch, D latch, J - K Flipflop, D Flipflop, master/slave flip-flop, edge triggered J-K flip-flop with asynchronous inputs, T flip-flops. Excitation tables, Characteristic tables, Characteristic equations

Unit-V **(8Contact hours)**

Frequency division and counting. Design and analysis of synchronous counters, asynchronous counters. State diagrams for D-flipflop, T-Flip flop, J-K Flip flop, Mealy machines and Moore machines.

Unit-VI **(8Contact hours)**

Design and display of mobile number using synchronous and asynchronous counters, design and display of digital clock using synchronous and asynchronous counters.

Learning Resources:

Text books:

1. Ronald J Tocci, Neal S.Widmer, Gregory L.Moss,'Digital systems' Pearson 10th edition.
2. Stephen Brown, Zvonko Vranesic,'Fundamentals of Digital Logic with Verilog Design', TMH, 2nd edition

Reference books

1. John F.Wakerly, 'Digital Design' , Pearson 4th edition

Web Resources

1. Prof. Shankar Balachandran, NPTEL-IIT Madras, *'Digital Circuits & Systems'*
   URL: https://nptel.ac.in/courses/117106114/
2. Prof. S Srinivasan, NPTEL-IIT Madras, 'Digital Circuits and Systems'
   URL: https://nptel.ac.in/courses/117106086/

Course Outcomes

At the end of the course, the student will be able to

CO 1	Apply the knowledge of simplification in obtaining optimal digital circuits
CO 2	Employ Boolean algebra to describe the function of logic circuits
CO 3	Design circuits which represent digital logic expressions. Specifically, design a gate-level digital circuit to implement a given Boolean function
CO 4	Study and examine the SSI, MSI, LSI and Programmable elements
CO 5	Analyse the operation of synchronous and asynchronous state machines

CO 6	Design any combinational or sequential digital circuits to meet the given specifications
CO 6	Analyse any digital circuit and to debug such circuit
CO 7	Prototype a real time application on EDA tool

Assessment Method

Assessment Tool	Weekly tests/Assignments (In semester)	Monthly tests (In semester)	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

****

Course code	Course name	Course category	L-T-P	Credits
20CS2101	Design & Analysis of Algorithms	PCC	3-1-0	4

Course Learning Objectives:

5. Interpret the fundamental needs of algorithms in problem solving
6. Classify the different algorithm design techniques for problem solving
7. Develop algorithms for various computing problems
8. Analyze the time and space complexity of various algorithms

Course Content

Unit I (8 Contact Hours)

Algorithm, Pseudo code for expressing algorithms, Performance Analysis-Space complexity, Time complexity, Asymptotic Notation- Big oh notation, Omega notation, Theta notation and Little oh notation, Probabilistic analysis, Amortized analysis. Analysis of Insertion Sort & Heap Sort.

Unit II (10 Contact Hours)

Divide and conquer: General Method, solving of recurrence relations – Substitution Method, Recursion Tree Method, Masters Method, applications-Binary search, Quick sort, Merge sort, Strassen's matrix multiplication,

Unit III (10 Contact Hours)

Greedy method: General method, applications-Job sequencing with deadlines, 0/1 knapsack problem, Minimum cost spanning trees, Disjoint Sets- disjoint set operations, union and find algorithms, spanning trees, connected components and bi-connected components, Single source shortest path problem.

Unit IV (10 Contact Hours)

Dynamic Programming: General method, applications-Matrix chain multiplication, Optimal binary search trees, 0/1 knapsack problem, All pairs shortest path problem, Travelling sales person problem, Reliability design.

Unit V (12 Contact Hours)

Backtracking: General method, applications-n-queen problem, sum of subsets problem, graph coloring, Hamiltonian cycles.

Branch and Bound: General method, applications - Travelling sales person problem, 0/1 knapsack problem- LC Branch and Bound solution, FIFO Branch and Bound solution.

Unit VI **(10 Contact Hours)**

String Matching: Naive string matching, Tries, Rabin Karp Algorithm, KMP algorithm, Boyer moore Algorithm.

NP-Hard and NP-Complete problems: Basic concepts, non deterministic algorithms, NP - Hard and NP Complete classes, Reducibility.

Learning resources

Text Books:

- Thomas H.Cormen, Charles E. Leiserson, Ronald L.Rivest, Clifford Stein , "Introduction to Algorithms".

Reference Books:

3. SatrajSahni and Rajasekharam,*"Fundamentals of Computer Algorithms",*
4. Galgotia publications pvt. Ltd.
5. ParagHimanshu Dave, HimanshuBhalchandraDave,*"Design and Analysis algorithms",*Publisher: Pearson.
6. R.C.T.Lee, S.S.Tseng, R.C.Chang and T.Tsai, McGraw Hill "*Introduction to Design and Analysis of Algorithms A strategic approach"*
7. Allen Weiss "*Data structures and Algorithm Analysis in C++",* Second edition,
8. Pearson education.
9. Aho, Ullman and Hopcroft "*Design and Analysis of algorithms*" Pearson education.

Web Resources

1.https://www.oreilly.com/library/view/design-and-analysis/9788177585957/
2.https://www.tutorialspoint.com/design_and_analysis_of_algorithms
3.https://www.amazon.in/Design-Analysis-Algorithms-V-Muniswamy/dp/9380026730

Course outcomes: At the end of the course, the student will be able to

CO 1	Synthesize efficient **algorithms** in common engineering **design** situations.
CO 2	Major techniques for **algorithm design** and **analysis** are introduced through the study of various **algorithms**.
CO 3	Apply design principles and concepts to algorithm design
CO 4	Have the mathematical foundation in analysis of algorithms
CO 5	Understand different algorithmic design strategies
CO 6	Analyze the efficiency of algorithms using time and space complexity theory

For Theory courses only:

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

****

Course code	Course name	Course Category	L-T-P	Credits
20CS2102	Database Management Systems	PCC	3-0-0	3

Course Learning Objectives:

1. To Understand the role of a database management system in an organization.
2. To Understand the basics of ER Diagram, Relational model, Relational Algebra and Relational Calculus.
1. To Understand basic database concepts, including the structure and operation of the relational data model.
2. To Construct simple and moderately advanced database queries using Structured Query Language (SQL).
3. To Understand and successfully apply logical database design principles, including E-R diagrams and database normalization.
4. To Understand the concept of a database transaction and related database facilities and indexing techniques.

Course Content:

Unit I (8Contacts hours)

Introduction to database systems, File System vs. Database Systems, Database system structure, Views of data in a database system, Data models and Database languages.

Introduction to Entity-Relationship data model, Elements of an ER model, Constructing ER diagrams, Modelling of constraints, Reduction of ER diagrams to relational tables.

Unit II (6 Contact hours)

Basics of relational model, ER diagrams to relational design, Relational algebra: Simple operations and extended operations, writing relational algebra expressions for queries, Introduction to tuple relational calculus and writing basic queries using tuple calculus

Unit III (9 Contact Hours)

Basic structure of SQL queries, Writing simple queries, Complex queries and nested Subqueries in SQL, Aggregate functions in SQL, Effect of NULL values on result, Defining a Relational Schema, View definitions and constraints, types of keys.

Unit IV (7 Contact hours)

Features of Good Relational Designs, Atomic Domains and First Normal Form, Problems encountered in bad schema design, Motivation for normal forms, Dependency theory-functional dependencies, Armstrong's Axioms for FD, Closure of a set of FD's, Minimal Cover, Definition of 1NF, 2NF, 3NF and BCNF, Decomposition and desirable properties of them.

Unit V (6Contact hours)

Storing data in disk and files and the memory hierarchy, RAID, File organization and indexes, ISAM Tree, B+ Tree, Linear Hashing and Extendible Hashing

Unit VI (9 Hours)

Transaction concept, ACID properties, Concurrency in a DBMS, Serializability and Recoverability, Concurrency control Protocols (lock-based and time-stamp based)

Text Books

A. Silberschatatz, H. F. Korth and S. Sudarshan, Database System Concepts, 5/e, McGraw Hill,2006

R. Ramakrishnan and J. Gehrke, Database System Concepts, 3/e, McGraw Hill, 2003

Wilfried Lemahieu, Seppe Vanden Broucke and Bart baesens Principles of Database Management Systems, 1/e Cambridge 2018

Reference Books

Ramez Elmasri, Shamkant B. Navathe , Fundamentals of Database (7th Edition), Paperback, 2007
Theorey T J, Database Modeling & Design, 2/e, Morgan Kaufmann Publishers, 1994.
H. GarciaMolina, J. D. Ullman and J. Widom, Database Systems The Complete Book, 1/e, Pearson Education, 2007

Web resources:

Department of CS&E, IIT M, "Introduction to Database Sytems and Design", https://nptel.ac.in/courses/106106095/
Indian Institute of Technology, Kharagpur, " Database Management Systems", https://nptel.ac.in/courses/106105175/

Course outcomes: At the end of the course, the student will be able to

CO 1	Demonstrate the basic elements of a relational database management system,
CO 2	Ability to identify the data models for relevant problems.
CO 3	Ability to design entity relationship and convert entity relationship diagrams into RDBMS and formulate SQL queries on the respect data.
CO 4	Apply normalization for the development of application software
CO 5	Ability to learn about Disk Management, Buffer management
CO 6	Ability learn about transaction management

For Theory courses only:

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

Course code	**Course name**	**Course category**	**L-T-P**	**Credits**
20CS2181	Design & Analysis of Algorithms Lab	PCC	0-0-3	1.5

Course Learning Objective:

1. This practical course should enable the students to
2. Lear how to analyze a problem and design the solution for the problem
3. Design and Implement efficient algorithms for a specified application
4. Strengthen the ability to identify and apply the suitable algorithm for a given real world problem.
5. To understand various graph algorithms

List of Experiments:

1. Lab No 1: Implementation and Analysis of Sorting Algorithms – Quick Sort, Merge Sort & Heap Sort
2. Lab No 2: Warshalls Algorithms – Applying to Topological Ordering of vertices in a given digraph and computing the transitive closure of given directed graph
3. Lab No 3: Implement 0/1 Knapsack Problem using Dynamic Programming

4. Lab No 4: Shortest Paths Algorithms : All Pair Shortest Path algorithms – Floyds Algorithm and other algorithms
5. Lab No 5: Implement any scheme to find the optimal solution for the Travelling Salesman Problem
6. Lab No 6: Implement Minimum Spanning Tree Algorithms – Prims Algorithms and Kruskal Algorithm
7. Lab No 7: Single Source Shortest Path Algorithms and other Graph Algorithms like connected components
8. Lab No 8: Implement the Sum of Subsets Problem
9. Lab No 9: Implementation of any scheme to solve the SUDOKU puzzle
10. Lab No 10: Implement N Queens Problem using the Back Trackin

Course Outcomes

At the end of the course, the student will be able to

CO 1	To analyze asymptotic notation and worst, average and best case analysis using suitable mathematical tools.
CO 2	To design efficient algorithms for computational problems using appropriate algorithmic paradigm.
CO 3	To understand different graph algorithms and traversal problems.
CO 4	To analyze the complexity of different class of problems.
CO 5	To explain the role of randomization and approximation in computation

Assessment Method

Course Nature		**Practical**		
Assessment Method				
Assessment Tool	Experiments	Record	Viva-Voce/ Quiz/MCQ/Lab project	Total
Weightage (%)	25%	5%	10%	40%
End Semester Examination weightage (%)				60%

Course Nature		**Practical**		
Assessment Method				
Assessment Tool	Experiments	Record	Viva-Voce/ Quiz/MCQ/Lab project	Total
Weightage (%)	25%	5%	10%	40%
End Semester Examination weightage (%)				60%

****

Course Code	**Course Name**	**Course Category**	**L-T-P**	**Credits**
20EC2180	Digital Logic Design Lab	**ESC**	**0-0-3**	**1.5**

Course Learning Objectives:

1. To expose to the concept of Digital knowledge and its applications
2. To understand Combinational and Sequential circuits

3. To design a prototype digital logic design

List of Experiments

1. Familiarization with logic gate IC's and Arduino kits
2. Design of code converters and comparators (8-bit) on bread board
3. Adder related experiments: Half adder , full adder , half subtractor, full subtractor , ripple carry adder, BCD adder, carry look ahead adder using IC
4. Design of a binary multiplier and displaying its inputs and outputs on seven segment display unit
5. Familiarization with multiplexer, decoder, encoder. Design of Half adder, full adder, magnitude comparator and other examples using above familiarized components
6. Bi-stable multi-vibrator design. Design and verification of SR,JK,D,T latch/flip-flops. Verification and elimination of Race Around Condition
7. Flip-flop conversions and Design of frequency dividers
8. Design of synchronous counters (Up and Down) and displaying result on seven segment display unit
9. Design of Mod $n \leq 2^n$ counter design(total 8 states, design of mod6 and mod7 with clear
10. Design and IC verification of Decade counter
11. Cascading of counters
12. Synchronous counter design and displaying result on seven segment display unit
13. Random sequence
14. Ring counter/Johnson counter
15. Verification and analysis of ALU IC
16. Design of a digital clock in synchronous state machine design and in asynchronous state machine design
17. Design and submission of term project

Note:

1. All the above experiments (except few exceptional cases) are to be implemented on Arduino kits also.
2. It is mandatory to perform experiment on any one of the EDA Tools before the experiment is done on hardware. All experiments must be unique, design specifications should not be common in the lab

Course outcomes:

After the completion of this Laboratory course, the student will be able to

CO 1	Understand the implementation of discrete digital components
CO 2	Utilize the ICs of Decoder, Multiplexer, Seven segment display unit in combination circuit design
CO 3	Utilize the ICs of suitable Flipflops in sequential circuit design
CO 4	Utilize the Programmable Logic devices in digital design
CO 5	Understand the concepts of setup time, hold time, propagation delays
CO 6	Design circuits with optimal features of Area, Power and delay
CO 7	Design and implement prototypes of complete digital systems

Assessment Method

Assessment Tool	Experiment	Report/Viva-Voce/ Quiz/MCQ/Lab	Total

	s	project	
Weightage (%)	25%	15%	40%
End Semester Examination weightage (%)			60%

Course Nature	**Practical**			
Assessment Method				
Assessment Tool	Experiments	Record	Viva-Voce/ Quiz/MCQ/Lab project	Total
Weightage (%)	25%	5%	10%	40%
End Semester Examination weightage (%)				60%

Course code	**Course name**	**Course Category**	**L-T-P**	**Credits**
20CS2182	**Database Management Systems Laboratory**	**PCC**	**0-0-3**	**1.5**

Course Objectives:

1. Analyze the problem and identify the Entities and Relationships, keys for given database.
2. Design, develop and query a database.
3. Able to construct queries and maintain a simple database using MySQL.
4. Normalization of data present in database tables.
5. Develop triggers programs using PL/SQL.

List of Experiments:

1. Designing the Database through Identifying Entities, Relationship Attributes.

MySQL

1. Queries to facilitate acquaintance of Built-In Functions, String Functions, Numeric Functions,
2. Queries to facilitate acquaintance of Date Functions and Conversion Functions.
3. Queries for Creating, Dropping, and Altering Tables
4. Queries using operators in SQL
5. Queries to Retrieve and Change Data: Select, Insert, Delete, and Update
6. Queries using Group By, Order By, and Having Clauses
7. Queries on Controlling Data: Commit, Rollback, and Save point
8. Queries for creating Views, and Constraints
9. Queries on Joins (Outer and Inner joins)
10. Queries on Correlated Sub-Queries

PL/SQL

1. Write a PL/SQL Code using Basic Variable, Anchored Declarations, and Usage of Assignment Operation
2. Write a PL/SQL block using SQL and Control Structures in PL/SQL
3. Write a PL/SQL Code using Cursors, Exceptions and Composite Data Types
4. Write a PL/SQL Code using Procedures, Functions, and Packages FORMS

Course Outcomes:

After completing this course the student must demonstrate the knowledge and ability to:

CO 1	Identify the entities, attributes, relationships, keys for given database.
CO 2	Design a database schema for given problem.
CO 3	Formulate queries using MySQL DML, DDL commands.

CO 4	Formulate SQL queries using constraints and set comparison operators.
CO 5	Apply the normalization techniques for development of application software to realistic problems.
CO 6	Develop PL/SQL programs using triggers, procedures
CO 7	Ability to design and implement given case study.

Course Nature		**Practical**		
Assessment Method				
Assessment Tool	Experiments	Record	Viva-Voce/ Quiz/MCQ/Lab project	Total
Weightage (%)	25%	5%	10%	40%
End Semester Examination weightage (%)				60%

ENGINEERING SECOND YEAR: SEMESTER-2

Course code	Course name	Course Category	L-T-P	Credits
20BM2202	**Introduction to Operation Research**	**HSC**	**3-0-0**	**3**

Course Learning Objective:

1. The objective of this course is to provide the exposure to the major tools and techniques of Operations Research.
2. To create awareness and appreciation about the applications of Operations Research in the functional areas of management.
3. To provide a formal quantitative approach to problem solving and an intuition about situations where such an approach is appropriate.
4. To introduce some widely- used mathematical models.
5. The understanding of these models will allow the students to communicate with persons who run them and to evaluate the results they present.
6. To provide a tool that the students can use to solve management problems

Course Contents:

Unit I: (6 hours)

Decision making, Development of OR, An overview and scope of Operations Research Application of OR

Unit II: (9 hours)

Linear Programming Problems (LPP), Introduction to Linear Programming (LP) Illustration of LP Problems. Formulation exercises on LP Problem, Graphical Method of solving LPP Simplex Method, Unboundedness, Multiple Optimum Solutions, Degeneracy and Cycling Problems.

UNIT III: (9 hours)

Artificial Variables : Big-M Method, Sensitivity Analysis, Duality Problems, Economic Interpretation of Simplex Tableau Computer Software for Solving LPP.

UNIT IV: **(9 hours)**

Formulation of Transportation Problems**,** Sensitivity Analysis in Transportation Problems**,** Assignment Problems.

UNIT V: **(7 hours)**

Elements of queuing models, Poisson arrival and exponential service time distributions, M/M/1 Queue; Finite population models. Queuing cost models, Applications.

UNIT VI: **(5 hours)**

Introduction of Costs, Deterministic and Stochastic models.

Learning Recourses:

Text Book:

1. Taha H.A., *Operation Research- An Introduction*, PHI, (2008)

Reference Books:

7. Ravindran, Phillips & Solberg, *Operations research*, John Wiley, Singapore, (2007)
8. Richard Levin & David Rubin, *Quantitative approaches to Management*, Mc GrawHill International, (1992).
9. Hillier & Lieberman, *Operation Research*, Addison Wesley, (1974)
10. Hadley G., *Linear Programming*, Addison-Wesley, (1962).

Web resources:

1. Introduction to Operation Research, Prof. G.S, Srinivasan, IIT Madras
https://nptel.ac.in/courses/110106062/

Course outcomes: At the end of the course, the student will be able to

CO 1	A student will be able to understand basic of operation research and its multipurpose uses
CO 2	It will provide application facilities of concepts.
CO 3	Students can be able to explore different uses of linear programming with its advance technique.
CO 4	It will enable to take decision regarding assignment, transportation and queuing related issues
CO 5	This will provide alternative techniques with its effective uses
CO 6	It will have a overall impact on the decision taking related to various type of real time issues.

For Theory courses only:

Course Nature		**Theory**		
Assessment Method				
Assessment	Weekly tests	Monthly tests	End Semester Test	Total

Tool				
Weightage (%)	10%	30%	60%	100%

****

Course code	Course name	Course Category	L-T-P	Credits
20CS2201	**COMPUTER ORGANIZATION AND ARCHITECTURE**	**PCC**	**3-0-0**	**3**

Course Learning Objectives:

1. To conceptualize the basics of organizational and architectural issues of a digital computer.
2. To analyze performance issues in processor and memory design of a digital computer.
3. To understand various data transfer techniques in digital computer.
4. To analyze processor performance improvement using instruction level parallelism

Course content:

UNIT-I

Basic Functional blocks of a computer: CPU, memory, input -output subsystems, control unit.

Data Representation: Number systems, signed number representation, fixed and floating point representations, character representation.

UNIT-II

ALU: Computer Integer Arithmetic: addition, subtraction, multiplication, division, floating point arithmetic: Addition, subtraction, multiplication, division.

Instruction set architecture of a CPU registers, instruction execution cycle, RTL interpretation of instructions, addressing modes, instruction set. RISC and CISC architecture. Case study instruction sets of some common CPUs.

UNIT-III

CPU control unit design: Introduction to CPU design, Processor Organization, Execution of Complete Execution, Design of Control Unit: hardwired and micro-programmed control, Case study design of a simple hypothetical CPU.

UNIT-IV

Memory system design: Concept of memory: Memory hierarchy, SRAM vs DRAM ,Internal organization of memory chips , cache memory: Mapping functions, replacement algorithms, Memory management, virtual memory.

UNIT-V

Input -output subsystems, I/O transfers: programmed I/O, interrupt driven and DMA.

I/O Buses, Peripheral devices and their characteristics, Disk Performance

UNIT-VI

Performance enhancement techniques: Pipelining: Basic concepts of pipelining,

Through put and speedup, pipeline hazards.

Parallel processing: Introduction to parallel processing, Introduction to Network, Cache coherence

Text Books:

V. C. Hamacher, Z. G. Vranesic and S. G. Zaky, *"Computer Organization,"* 5/e, McGraw Hill, 2002.

William Stallings, *"Computer Organization and Architecture"*: Designing for Performance, 8/e, Pearson Education India. 2010.

Morris Mano, " Computer System Architecture", Pearson Education India, Third edition.

References:

A. S. Tanenbaum, "*Structured Computer Organization"*, 5/e, Prentice Hall of India, 2009.

D. A. Patterson and J. L. Hennessy, *"Computer Organization and Design,"* 4/e, Morgan Kaufmann, 2008.

J. L. Hennessy and D. A. Patterson," *Computer Architecture: A Quantitative Approach"*,4/e, Morgan Kaufmann, 2006.

D. V. Hall, "*Microprocessors and Interfacing"*, 2/e, McGraw Hall, 2006

*" 8086 Assembler Tutorial for Beginners "*By Prof. Emerson GiovaniCarati.

Web referneces:

https://en.wikibooks.org/wiki/IB/Group_4/Computer_Science/Computer_Organisation

http://www.cs.uwm.edu/classes/cs458/Lecture/HTML/ch05.html

http://www.cse.iitm.ac.in/~vplab/courses/comp_org.htm

OA IIT-Guwahati Pdf

Course outcomes: At the end of the course, the student will be able to

CO1	Understand the basic organization of computer and different instruction formats and addressing modes.
CO2	Analyze the concept of pipelining, segment registers and pin diagram of CPU.
CO3	Understand and analyze various issues related to memory hierarchy.
CO4	Evaluate various modes of data transfer between CPU and I/O devices.
CO5	Examine various inter connection structures of multi processors.

For Theory courses only:

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

****

Course code	Course name	Course Category	L-T-P	Credits
20CS2202	Data Science with Python	PCC	3-0-0	3

Unit – I : Python Basics for Data Science

Introduction to Python, Types, Expressions & Variables, String Operations, Lists & Tuples, Sets, Dictionaries, Conditions & Branching, Loops, Functions, Objects & Classes

Unit – II: Working with Data in Python

File Operations, Regular Expressions, Pandas, NumPys, Web Scraping

Unit – III: Data Processing

Importing DataSets; Cleaning & Preparing Data – Handling Missing Values, Data Formatting, Binning;Summarizing the Data Frame – Descriptive Statistics, Grouping, ANOVA, Correlation

Unit – IV: Data Analysis

Model Development : Simple & Multiple Linear Regression, Model Evaluation using Visualization, Polynomial Regression; Model Evaluation – Overfitting, Underfitting, Model Selection, Ridge Regression, Model Refinement

Unit – V: Data Visualization

Introduction to Visualization Tools – Matplotlib, Line Plots, Area Plots, Histograms, Bar Charts, Pie Charts, Box Plots, Scatter Plots, Bubble Plots; Advance Visualization Tools – Waffle Charts, Word Clouds, Seaborn and Regression Plots; Creating Maps & Visualizing Geospatial Data – Folium, Maps with Markers, Choropleth Maps

Unit – VI: Machine Learning using Python

Introduction to Machine Learning – Supervised vs Unsupervised Learning, Python Libraries for Machine Learning; Regression; Classification; Unsupervised Learning; Recommender Systems

Learning resources:

Text Book:

1) Jake VanderPlas, Python Data Science Handbook - Essential Tools for Working with Data, o'reilly publications.

Online Course Reference:

https://www.edx.org/professional-certificate/python-data-science

****

Course Code	Course Name	Course Category	L-T-P	Credits
20CS2204	Compiler Design	PCC	3-0-0	3

Course Learning Objectives:

1. To implement the concept learned in automata theory and languages to the field of Computer Science.
2. Analyze the basic steps involved in converting a source language to target code.
3. Understands the concepts of parsers and can write solutions for various grammars by using tools, and also analyzes different storage techniques, error recovery strategies
4. Gain the knowledge to write a compiler program or can able to build a compiler.

Course Content:

Unit I

Introduction to Compilers (6 Contact hours)

Introduction to compilers, Phases of compiler, Lexical Analyzer, The role of the lexical analyzer, input buffering, specification of tokens, Recognition of tokens.

Unit II

Syntax Analysis -I (9 Contact hours)

Role of the parser, writing grammars and context free grammars, Top down parsing, Brute-force approach, Recursive descent parsing, Predictive parsing, FIRST and FOLLOW constructs.

Unit III

Syntax Analysis -II (8 Contact hours)

Bottom-up parsing, shift-reduce parsing, operator precedence parsing, LR parsers, SLR parser, canonical LR parser, LALR parser.

Unit IV

Semantic Analysis (8 Contact hours)

Syntax directed translations, applications of syntax directed translations, Syntax directed definitions, construction of syntax tree, Bottom-up evaluation of S-attributed definitions, L-attributed definitions.

Unit V

Intermediate Code Generation and Code Optimization (8 Contact hours)

Intermediate languages, Declarations, Assignment statements, Boolean Expressions, case statements, back patching, Procedure calls, Principal sources of optimization, optimization of basic blocks, DAG representation of basic blocks, flow graphs.

Unit VI

Code generation (6 Contact hours)

Issues in the design of code generator, the target machine, run time storage management, peephole optimization.

Learning resources

Text book:

1. Alfred V Aho, Monica S Lam, Ravi Sethi, Jeffrey D Ullman, *"Compilers: Principles Techniques & Tools"*, Pearson Education,2nd Edition 2013.

Reference Books:

1. Kenneth C Louden, *"Compiler Construction: Principles and Practice*", Cengage Learning. Lex &Yacc, John R Levine, Oreilly Publishers.
2. Keith D Cooper & Linda Tarezon, *"Engineering a Compiler*", Morgan Kafman, Second edition. Lex &Yacc, John R Levine, Tony Mason, Doug Brown, Shroff Publishers.
3. Muchnik, "*Advanced Compiler Design and Implementation*", Kauffman(1998)

Course outcomes:

CO	Identify the basic concepts needed for the development of a compiler

1	
CO 2	Analyze the various phases and Tools of a Compiler
CO 3	Describe the differences between Top-Down and Bottom-Up Parsers and apply parsing methods for various grammars.
CO 4	Compare and Contrast Symbol table organization for Block Structured and Non-Block Structured languages.
CO 5	Analyze the concepts involved in Intermediate, Code Generation and Code Optimization Process.
CO 6	Recognize the various types of errors and error recovery strategies in phases of Compilation

Assessment Method:

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests (In semester)	Monthly tests (In semester)	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

Course Code	Course Name	Course Category	L-T-P	Credits
20CS2203	**Web Technologies**	**PCC**	**3-0-0**	**3**

Course Learning Objectives:

1. To demonstrate basic skills in analysing the usability of a web site.
2. To identify how to plan user research related to web design.
3. To Learn how to design, add client side script and publish web page
4. To Learn about server side programming and deploy the app into a server
5. To Learn about storing the data into SQL and NoSQL
6. To Learn about Front-End Web UI Frameworks and GIT repository Tools
7. To learn the language of the Web: jQuery Frontend design and Bootstrap
8. To learn the language of Web: NodeJS

UNIT I (10 Contact hours)

Introduction to Web World: Recap on HTML, inserting Frames and frame sets, inserting hyperlinks, lists, tables and images,

JavaScript : Client side scripting with JavaScript, variables, functions, conditions, loops and repetition, Pop up boxes, objects,HTML DOM and web, Browser environments, form validation, Events and Event Listeners

UNIT II (10 Contact hours)

Server Programming: PHP basics: PHP Syntax, variables, constants, Data Types, Strings, Conditional and Control Structures. PHP GET and POST. PHP Advanced: include files, File system, parsing directories, file upload and download, Sessions, Form handling, JSON Parsing

UNIT-III (10 Contact hours)

Database Connectivity: Introduction to SQL: Connect, create database, create table, insert, prepared statements. Use of NoSQL: Introduction to NoSQL, Difference between SQL and NoSQL, Types of NoSQL Databases, Query mechanism tools for NoSQL.

Authentication: Google OAuth: Basic Steps. Access to Google APIs: For Server-side Web apps, for Java Script Web apps, for Mobile & Desktop apps

UNIT IV (10 Contact hours)

Front-End Web UI Frameworks and Tools: Bootstrap, Full-Stack Web Development, Setting up Git, Basic Git commands, Online Git Repositories, Node.js and NPM, Front-end Web UI Frameworks.

jQuery: Introduction, Selectors, Attributes, Event Handlers, Style Methods, Traversing the DOM, Effects, and Introduction to jQuery Plugins

UNIT V **(7Contact hours)**

Bootstrap: Introduction to Bootstrap, Responsive Design, Bootstrap Grid system, Navigation and Navigation Bar, Icon Fonts, User Inputs, Bootstrap CSS Components, Bootstrap and JavaScript Components, Bootstrap and JQuery, Building and Deployment, NPM Scripts, Task Runners

UNIT VI **(7Contact hours)**

NodeJs: Introduction, Environment Setup, First Application, REPL Terminal, Native Package Manager(NPM), Web Module, Express Framework, RESTFul API

Learning resources:

Text Books

1. Ralph Moseley and M. T. Savaliya, WileyIndia "*Developing Web Applications*"
2. Jeffrey C.Jackson,"*Web Technologies--A Computer Science Perspective*", PearsonEducation,
3. Dreamtech Press "*Web TechnologiesBlack Book*," HTML 5,
4. Web Technologies: HTML, JAVASCRIPT, PHP, JAVA, JSP, XML and AJAX, Black Book Kindle Edition by Kogent Learning Solutions Inc. (Author)
5. O'Reilly - Head First Servlets and JSP, 2nd Edition
6. Node.js Web Development: Create real-time server-side applications with this practical, step-by-step guide, 3rd Edition
7. Bootstrap: Responsive Web Development Book by Jake Spurlock

Reference Books

1. Joel Sklar, Cengage *" Web Design*", Learning
2. Robert. W. Sebesta, "*Programming the World Wide Web*", Fourth Edition, PearsonEducation
3. P.J. Deitel& H.M. Deitel "*Internet and World Wide Web How to program*", Pearson

Web Resources

https://www.w3schools.com/
https://www.tutorialspoint.com/web_development_tutorials.htm
https://html.com/
https://www.coursera.org/learn/bootstrap-4
https://www.tutorialspoint.com/jquery/index.htm
https://www.tutorialspoint.com/nodejs/
http://www.ntu.edu.sg/home/ehchua/programming/java/javaservlets.html
http://wiki.lib.sun.ac.za/images/0/07/Bootstrap-tutorial.pdf
https://media.readthedocs.org/pdf/htmlguide/latest/htmlguide.pdf

Course outcomes: At the end of the course, the student will be able to

CO 1	Learn how to design, add client side script and publish web page

CO 2	Learn how to write server side programming and deploy the app into a server.
CO 3	Learn how to store data into database and NoSQL.
CO 4	Learn about Front-End Web UI Frameworks and GIT repository Tools.
CO 5	Learn about responsive Web design.
CO 6	Learn about Package manager and Web modules.

****

Course code	Course Name	Course Category	L-T-P	Credits
20CS2281	**Computer Organization and Architecture Laboratory**	**PCC**	**0-0-3**	**1.5**

Course Learning Objective

1. To expose the students to the various key aspects of Computer Organization & Architecture.
2. To aquaint with various registers in the CPU and understand about the assembly language programming.

List of Experiments:

Lab No 1.a) Verification of Logic gates
b) Assembly language program to find largest number in an Array.

Lab No 2. a) Verification of Full-Adder and Full-Subtractor
b) Assembly language program to find smallest number in an array.

Lab No 3. a)Verification of Ripple Carry Adder and Carry-look-ahead adder.
b) Assembly language program for adding to two arrays
Lab No 4. a) Combinational Multipliers
b) Assembly language program to separate even and odd numbers from an array.
Lab No 5. a) Booth's Multiplier
b) Assembly language rogram to find prime numbers between a given range

Lab No 6. a) Wallace Tree Adder
b) Assembly language program to find factorial of the given number.

Lab No 7. a) Arithmetic Logic Unit
b) Assembly language program to find LCM.
Lab No 8. a) Verification of Registers and Counters
b) Assembly language program to find GCD.
Lab No 9. a) Memory Design
b) Assembly language program to search an element using linear search.

Lab No 10. a) Direct Mapped cache Design
b) Assembly language program to search an element using binary search.

Lab No 11. a) Associative cache Design

b) Assembly language program to sort numbers using bubble sort.

Course Outcomes

At the end of the course, the student will be able to

CO 1	Understand the basic logic gates
CO 2	Understand the full adder and full subtractor
CO 3	**Ripple Carry Adder** examine the behavior of the working module to understand how the carry ripples through the adder stages to design a ripple carry adder using full adders to mimic the behavior of the working module . **Carry Lookahead Adder** understand the behaviour of carry lookahead adder understand the concept of reducing computation time with respect of ripple carry adder by using carry generate and propagate functions
CO4	**Combinational Multipliers** understand the behaviour of combinational multiplier . understand the scheme implemented for the multiplication. it can be designed by unrolling the multiplier loop instead of handling the carry out of partial product summation bit,the carry out can be sent to the next bit of the next step this scheme of handling the carry is called *carry save addition*
CO 5	**Booth's Multiplier** Understand the behaviour of Booth's multiplication**.** Design Booth's multiplier with a controller and a datapath. This will also help in the learning of control unit design as a finite state machine Understand the advantages of Booth's multiplier It can handle signed integers in 2's complement notion It decreases the number of addition and subtraction It requires less hardware than combinational multiplier It is faster than straightforward sequential multiplier
CO 6	**Wallace Tree Adder** Understand the behaviour of wallace tree**.** understand the concept of reducing gate delay by using tree of adders instead of using cascaded full adders
CO 7	**Arithmetic Logic Unit** Understand the behaviourof arithmetic logic unit. Design an arithmetic logic unit for given parameter**.**
CO8	**Registers** to understand the shifting of data to examine the behavior of different modes of data input and data output(serial-in

	serial-out, serial-in parallel-out, parallel-in serial out,parallel-in parallel-out) to make use of shift register in data transfer developing skills in the designing and testing of sequential logic circuits developing skills in analysing timing signals. **Counters** understand the concept of counting upto certain limiting value and returning back to the start state from final state understand the generation of timing sequences to control operations in a digital system develop skills in the design and testing of counters for given timing sequences develop skills in generating timing signals .
CO 9	**Memory Design** Understand the **behavior of memory.** **D**esign memory for given parameter**.**
CO 10	**Direct Mapped Cache Design** Understand the behavior of direct mapped cache from working module Design a direct mapped cache for given parameters.
CO 11	**Associative Cache Design** Understand the behavior of associative cache. Designs a associative cache for given parameters**.**
Understand and develop Assembly Language Programs	

Assessment Method

Assessment Tool	Experiments	Record/Viva-Voce/ Quiz/MCQ/Lab project	Total
Weightage (%)	25%	15%	40%
End Semester Examination weightage (%)			60%

****

Course code	Course name	Course Category	L-T-P	Credits
20CS2282	**Data Science using Python Lab**		**0-0-3**	**1.5**

Lab:

Experiment 1:

a) **Python Basics:** Your first program, Types Expressions and Variables String Operations

b) **Python Data Structures:** Lists and Tuples Sets,and Dictionaries

c) **Python Programming Fundamentals:** Conditions and Branching Loops, Functions, Objects and Classes

d) **Working with Data in Python:** Reading files with open, Writing files with open, Loading data with Pandas, Working with and Saving data with Pandas

e) **Working with Numpy Arrays:** Numpy 1d Arrays, Numpy 2d Arrays

Experimen 2:

a) **Importing Datasets:** Learning Objectives, Understanding the Domain, Understanding the Dataset, Python package for data science, Importing and Exporting Data in Python, Basic Insights from Datasets

b) **Cleaning and Preparing the Data:** Identify and Handle Missing Values, Data Formatting, Data Normalization Sets, Binning, Indicator variables

c) **Model Development:** Simple and Multiple Linear Regression, Model EvaluationUsingVisualization, Polynomial Regression and Pipelines, R-squared and MSE for In-Sample Evaluation, Prediction and Decision Making

d) **Summarizing the Data Frame:** Descriptive Statistics, Basic of Grouping, ANOVA, Correlation, More on Correlation

e) **Model Evaluation:** Model Evaluation, Over-fitting, Under-fitting and Model Selection, Ridge Regression, Grid Search, Model Refinement

Experiment 3:

a) **Introduction to Visualization Tools:** Introduction to Data Visualization,Introduction to Matplotlib, Basic Plotting with Matplotlib,Dataset on Immigration to Canada, Line Plots

b) **Basic Visualization Tools:** Area Plots,Histograms,Bar Charts

c) **Specialized Visualization Tools:** Pie Charts, Box Plots, Scatter Plots, Bubble Plots

d) **Advanced Visualization Tools:** Waffle Charts, Word Clouds,Seaborn and Regression Plots

Expeiment 4:

a) **Introduction to Machine Learning:** Applications of Machine Learning, Supervised vs Unsupervised Learning, Python libraries suitable for Machine Learning

b) **Regression:** Linear Regression, Non-linear Regression, Model evaluation methods

c) **Classification:** K-Nearest Neighbour, Decision Trees, Logistic Regression, Support Vector Machines,Model Evaluation

d) **Unsupervised Learning:**K-Means Clustering,Hierarchical Clusterin, Density-Based Clustering

Assessment Method

Assessment Tool	Experiments	Report/Viva-Voce/ Quiz/MCQ/Lab project	Total
Weightage (%)	25%	15%	40%
End Semester Examination weightage (%)			60%

****

Course code	Course name	Course Category	L-T-P	Credits
20CS2283	**Web Technologies Lab**	**PCC**	**0-0-3**	**1.5**

Course Learning Objective:

The course will enable the students to:

1. Demonstrate the ability to retrieve data from a database and present it in a web page.
2. Create web pages that meet accessibility needs of those with physical disabilities and apply the effects of CSS in web page creation.
3. Create effective scripts using JavaScript and jQuery to enhance the end user experience.
4. Demonstrate knowledge of introductory programming concepts.
5. Test, debug, and deploy web pages containing JavaScript and jQuery

6. learn to create pages common to all web applications, and implement the most frequently used components and classes provided by Bootstrap
7. Understand the JavaScript and technical concepts behind Node JS
8. Understand the Servlet programming and deploying application in Web server

List of Experiments:

1. Install and configure the IDE
2. Incorporating JavaScript on an HTML page, and how to link to an external .js file
3. Comparing JavaScript with jQuery for same tasks
4. Using major methods/events in jQuery
5. Using Plugins and local data storage
6. Implement Bootstrap in existing web sites
7. Common Bootstrap components and use Bootstrap themes
8. Setup a Node.js project using npm
9. Use the Node.js core modules

Course Outcomes

At the end of the course, the student will be able to

CO 1	and configure the IDE
CO 2	orating JavaScript on an HTML page, and how to link to an external .js file
CO 3	Comparing JavaScript with jQuery for same tasks
CO 4	major methods/events in jQuery
CO 5	Plugins and local data storage
CO 6	Implement Bootstrap in existing web sites
CO 7	ıon Bootstrap components and use Bootstrap themes
CO 8	a Node.js project using npm
CO 9	e Node.js core modules
CO 10	ng Servlet application and deploying application in web server

Assessment Method

Assessment Tool	Experiments	Report/Viva-Voce/ Quiz/MCQ/Lab project	Total
Weightage (%)	25%	15%	40%
End Semester Examination weightage (%)			60%

****

ENGINEERING THIRD YEAR : SEMSTER-I

Course Code	Course Name	Course Category	L-T-P	Credits
20CS3101	**Operating System**	**PCC**	**3-0-0**	**3**

Course Learning Objectives

1. To learn the fundamentals of Operating Systems.
2. To learn the mechanisms of OS to handle processes and threads and their communication

3. To learn the mechanisms involved in memory management in contemporary OS
4. To know the components and management aspects of concurrency management
5. To learn programmatically to implement simple OS mechanisms

Course Content:

Unit - I **(6 Contact Hours)**

Basics: Evaluation, definition, Operating System Functionalities, Types of Operating Systems, Computer Architecture support to Operating Systems: Kernel and user mode. Introduction to Systems calls.

Unit - II **(7 Contact Hours)**

Process Management: definition: Process and PCB, description, Life cycle, Process Scheduling: Preemptive and Non-Preemptive; (Round Robin, FIFO, SJF and priority based) Uniprocessor scheduling algorithms, Multiprocessor and Real-time scheduling algorithms

Unit - III **(8 Contact Hours)**

Process Synchronization - Peterson's Solution, Banker's Algorithm,, Semaphores, Critical Regions: Producer-consumer problems, Readers writers problem, dining Philosophers problem. Monitors

Unit - IV **(7 Contact Hours)**

Introduction to deadlocks, Resource allocations, Deadlock Conditions, Deadlock prevention, Deadlock Detection- safe and unsafe states, deadlock avoidance- Banker's algorithms, and Recovery.

Unit - V **(10 Contact Hours)**

Memory Management: Partitioning, Paging and Segmentation and space allocation; Page replacement algorithms, Analysis of page allocation policies - Working Set, Virtual memory, Demand Paging.

Unit – VI **(9 Contact Hours)**

File Systems And Secondary Storage Management: : Free space management: Contiguous, Sequential and Indexed Allocation, File system interface, File System implementation, Disk Scheduling, Device drivers - block and character devices, streams, Character and Block device switch tables,

Learning Resources

Text book:

1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne,"*Operating System Concepts*", John Wiley & Sons Inc., 6th Edition.
6. William Stallings, "*Operating System: Internals and Design Principles*", Pearson, 5th Edition.

Reference Books:

1. Andrew S Tanenbaum, "*Modern Operating Systems*", Pearson Prentice Hall, 4th Edition.
2. Systmes D M Dhamdhere, *Operating Systems - System Programming and Operating*, Tata McGraw Hill
3. Gary Nutt, *Operating Systems: A Modern Perspective*, Addison Wesley, 2nd Edition.

Web resources:

1. PCP Bhattm, December 31 2009, *Operating Systems*, https://nptel.ac.in/courses/106108101/

Course outcomes: At the end of the course, the student will be able to

CO 1	Identify the structure of OS and basic architectural components involved in OS design.
CO 2	Explain the Mutual exclusion, Deadlock Handling Methods
CO 3	Design applications to simulate process scheduling and memory management algorithms.
CO 4	Differentiate the system functionalities in between old and modern OS
CO 5	Tell the need of protection and security in OS

For Theory courses only:

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

****

Course Code	Course Name	Course Category	L-T-P	Credits
20CS3102	**Computer Networks**	**PCC**	**3-0-0**	**3**

Course Learning Objectives:

1. Build an understanding of the fundamental concept of computer networking.
2. Formalize the student with the basic terminology of the computer networking.
3. Introduce the advanced networking concepts.
4. Preparing the students for entry Advanced courses in Computer Networking.

Course Content:

Unit - I **(7Contact hours)**

Introduction: Network Hardware, Network Software, References Models. Physical Layer-Guided medium and unguided medium , topologies.

Unit - II **(8 Contact hours)**

The Data Link Layer : Data link Layer Design Issues, Error Control.

Elementary Data Link Protocols, Sliding Window Protocols. The Medium Access Control Sublayer: The Channel allocation Problem, Multiple Access protocols, Ethernet - Ethernet Cabling, standard Ethernet , Switched Ethernet, Fast Ethernet.

Unit - III **(9 Contact hours)**

The Network Layer: Network Layer Design Issues, Routing Algorithms, Congestion Control Algorithms. Internetworking,subnetting, The Network Layer in the Internet.

Unit - IV **(7 Contact hours)**

The Transport Layer: The Transport Service, Elements of Transport Protocols, The Internet Transport Protocols: UDP, The Internet Transport Protocols: TCP.

Unit - V **(7 Contact hours)**

The Application Layer: DNS - The Domain Name System, Electronic Mail. The World Wide Web, file transfer protocol Multimedia.

Unit – VI **(7 Contact hours)**

Network Security: Cryptography, Symmetric - Key Algorithms, Public - Key Algorithms, Digital Signatures.

Learning resources

Text book:

1. Andrew S. Tanenbaum "*Computer Networks",* 4e, Pearson Education.
2. Behrouz A. Forouzan *"Data Communications and Networking"*, 4e , Tata McGraw Hill.
3. W. Stallings, "*Data and Computer Communication*", 8e, Pearson

Reference Books:

1. S. Kshev *"An Engineering Approach to Computer Networks "*, 2nd edition
2. W.A.Shay,Thomson *"Understanding Communications and Networks*",3rd edition,

Web resources:

1. https://www.tutorialspoint.com/computer_fundamentals/computer_networking.htm
2. https://www.geeksforgeeks.org/computer-network-tutorials/
3. https://nptel.ac.in/courses/106105081/

Course outcomes: At the end of the course, the student will be able to

CO 1	Independently understand basic Computer Network technology
CO 2	Identify the different types network topology and protocols
CO 3	Enumerate the OSI layers and TCP/IP.
CO 4	Explain the each layer functions.

For Theory courses only:

Course Nature	**Theory**

Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

****

Course Code	Course Name	Course Type	L-T-P	Credits
20CS3103	**Software Engineering**	**PCC**	**3-0-0**	**3**

Course Learning Objectives:

1. To recognize the emergence and importance of Software engineering
2. To identify the different phases in Software Development Life Cycle
3. To prepare the SRS(Software Requirement Specifications) Document for suitable product
4. To discuss the characteristics of good software design
5. To draw the UML Pattern designs for suitable design

6. To validate the product using various testing methods for producing quality software product.

Course Content:

Unit - I **(7 Contact Hours)**

Introduction: Introduction to Software Engineering, Exploratory style Vs. Modern style of development; Software Development Life Cycle; Process Models.

Unit - II **(8 Contact Hours)**

Software Project management: project Planning, estimation, **Software requirements and specification**: gathering, analysis, specification, characteristics, organization.

Unit – III **(9 Contact Hours)**

Software design: overview, characteristics of good design, function-oriented software design, object oriented design, UML, design patterns.

Unit –IV **(7 Contact Hours)**

Coding: Implementation, Coding Standard and Guidelines, review, Unit Testing; Verification and validation.

Unit – V **(7 Contact Hours)**

Testing: Integration and systems testing, Black box & White Box Testing, debugging techniques.

Unit – VI **(7 Contact Hours)**

Software Reliability And Quality Management: Software quality, SEI CMM and ISO- 9001, Reliability, Safety, Risk Analysis, computer-aided software engineering (CASE).

Learning Resources:

Text Book:

8. 1.Rajib Mall, *'Fundamentals of Software Engineering'*, PHI; Fourth edition (2014)
9. 2.Pressman, R.S., *'Software Engineering: A Practitioner's Approach'*, McGraw Hill, seventh edition, 2010.
10. Pankaj Jalote. *'An Integrated Approach to Software Engineering'*, 2nd edition, Narosa Publishing House

Reference Books:

4. Pressman, R.S., *'Software Engineering: A Practitioner's Approach'*, McGraw Hill, seventh edition, 2010.
5. Pankaj Jalote. *'An Integrated Approach to Software Engineering'*, 2nd edition, Narosa Publishing House
6. Bennett S., McRobb S. & Farmer R., *'Object Oriented Systems Analysis and Design using UML'*, Tata McGraw-Hill, second edition, 2004.
7. Summerville Ian, *'Software Engineering'*, Addison-Wesley, fifth edition, 2000
8. K.K.Agarwal, '*Software Engineering'*

Video Resources :

7. Primary Producer: NPTEL: *Prof. Sarda*, IIT Bombay, Publication Date: October 8, 2008, '*Introduction to Software Engineering'* URL **https://nptel.ac.in/courses/106101061/**

Web Resources:

1. https://www.tutorialspoint.com/software_engineering/
2. https://www.geeksforgeeks.org/software-engineering
3. https://onlinecourses.nptel.ac.in/noc18_cs43

Course Outcomes: At the end of the course the students will be able to

CO1	Describe the software engineering lifecycle by demonstrating competence in communication, planning, analysis, design, construction, and deployment
CO2	An ability to work in one or more significant application domain
CO3	To develop and deliver quality software by working as an individual and as part of a multidisciplinary team

CO4	Demonstrate an understanding of and apply current theories, models, and techniques that provide a basis for the software lifecycle
CO5	Deliver quality software products by possessing the leadership skills as an individual or contributing to the team development and demonstrating effective and modern working strategies by applying both communication and negotiation management skill.
CO6	Apply new software models, techniques and technologies to bring out innovative and novelistic solutions for the growth of the society in all aspects and evolving into their continuous professional development.

For Theory courses only:

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

Course code	**Course name**	**Course Category**	**L-T-P**	**Credits**
20CS3104	**Mathematical Foundations for Data Science**	**PCC**	**3-1-0**	**3**

Unit 1:

LINEAR ALGEBRA BASICS- Vector spaces and subspaces, basis and dimensions, linear transformation, four fundamental subspaces MATRIX THEORY- Norms and spaces, eigenvalues and eigenvectors, Special Matrices and their properties, least squared and minimum normed solutions.

Unit 2:

MATRIX DECOMPOSITION ALGORITHMS- SVD: Properties and applications, low rank approximations, Gram Schmidt process, polar decomposition

Unit 3:

DIMENSIONS REDUCTION ALGORITHMS and JCF- Principal component analysis, linear discriminant analysis, minimal polynomial and Jordan canonical form

Unt 4:

CALCULUS – Basic concepts of calculus: partial derivatives, gradient, directional derivatives, jacobian, hessian, convex sets, convex functions and its properties

Unit 5:

OPTIMIZATION – Unconstrained and Constrained optimization, Numerical optimization techniques for constrained and unconstrained optimization: Newton's method, Steepest descent method, Penalty function method

Unit 6:
PROBABILITY – Basic concepts of probability: conditional probability, Bayes' theorem, independence, theorem of total probability, expectation and variance, few discrete and continuous distributions, joint distributions and covariance.

BOOKS AND REFERENCES

1. W. Cheney, Analysis for Applied Mathematics. New York: Springer Science+Business Medias, 2001.
2. S. Axler, Linear Algebra Done Right (Third Edition). Springer International Publishing, 2015.
3. J. Nocedal and S. J. Wright, Numerical Optimization. New York: Springer Science+Business Media, 2006.
4. J. S. Rosenthal, A First Look at Rigorous Probability Theory (Second Edition). Singapore: World Scientific Publishing, 2006.

Online Resorces:
Online cource: Essential Mathematics for Machine Learning:
https://nptel.ac.in/courses/111/107/111107137/

****

Course Code	Course Name	Course Category	L-T-P	Credits
20CS3181	**Operating System Lab**	**PCC**	**0-0-3**	**1.5**

Career Objectives:
5. To understand the design aspects of operating system.
6. To study the process management concepts & Techniques.
7. To study the paging and segmentation concepts.
8. To familiarize students with the Linux environment

List of Experiments:
1. Introduction to system calls – implementation of open(), creat(),close(), write(), read(), lseek()
2. Implementation of fork (), wait (), exec() and exit () system calls
3. Write a program to simulate the following non-preemptive CPU scheduling algorithms to find turnaround time and waiting time.
   a) FCFS b) SJF
4. Write a program to simulate the following preemptive CPU scheduling algorithms to find turnaround time and waiting time.
   a) Round Robin b) Priority
5. Write a program to simulate producer-consumer problem using multi-threading.
6. Write a program to simulate Bankers algorithm for the purpose of deadlock avoidance.
7. Write a program to simulate the following contiguous memory allocation techniques
   a) Worst-fit b) Best-fit c) First-fit
8. Write a program to simulate paging technique of memory management.
9. Write a C program to simulate page replacement algorithms
   a) FIFO b) LRU
10. Write a C program to simulate disk scheduling algorithms
    a) FCFS b) SCAN c) C-SCAN
11. Study and practice of Unix/Linux general purpose utility command list

man,who,cat, cd, cp, ps, ls, mv, rm, mkdir, rmdir, echo, more, date, time, kill, history, chmod, chown, finger, pwd, cal, logout, shutdown.

12. Write a C program that makes a copy of a file using standard I/O, and system calls
13. Write a C program to emulate the UNIX **ls –l** command.
14. Write a C program that illustrates how to execute two commands concurrently with a command pipe.

    Ex: - l**s –l | sort**

15. a) Study of Bash shell, Bourne shell and C shell in Unix/Linux operating system.
    b) Study of Unix/Linux file system (tree structure).
    c) Study of .bashrc, /etc/bashrc and Environment variables.

At the end of this lab session, the student will be able

Course Outcomes:

CO 1	To use Unix utilities and perform basic shell control of the utilities
CO 2	To use the Unix file system and file access control.
CO 3	To use of an operating system to develop software
CO 4	To use Linux environment efficiently

Assessment Method

Assessment Tool	Experiments	Report/Viva-Voce/ Quiz/MCQ/Lab project	Total
Weightage (%)	25%	15%	40%
End Semester Examination weightage (%)			60%

****

Course Code	Course Name	Course Category	L-T-P	Credits
20CS3182	**Computer Networks Lab**	**PCC**	**0-0-3**	**1.5**

Course Learning Objectives:

1. To understand the principles of subnetting and types of sub netting
2. To Simulate the network communication and packet routing
3. To differentiate various subnet masking models
4. To analyze the various routing algorithms
5. To know the concept Virtual Lan's and Spanning ports

List of Experiments:

1. Lab No 1: Understanding and using of commands like ifconfig, netstat, ping, arp, telnet, ftp, finger, traceroute, whois etc..
2. Lab No 2: Understanding packet tracer and independent components in packet tracer
3. Lab No 3: Understanding the subnetting concept and types, Fixed Length subnet Masking (FLSM)and Variable length subnet masking (VLSM).
4. Lab No 4: To implement Static Routing using FLSM & VLSM.
5. Lab No 5: To configure DHCP as a pool router & Service Server.
6. Lab No 6: To implement Dynamic Routing using RIP version 1
7. Lab No 6: To implement Dynamic Routing using RIP version 2
8. Lab No 7: To implement Dynamic Routing using OSPF single area network
9. Lab No 8: To implement Dynamic Routing using OSPF multi area network

10. Lab No 9: To implement NAT(Network Address Translation) ,PAT(Port Address Translation)
11. Lab No 10: To implement VLAN
12. Lab No 11: Implementation of Socket program (Client server chat application using JAVA)
13. Lab No 12: Implementation FTP application to transfer multimedia

Course Outcomes

At the end of the course, the student will be able to

CO 1	Understand details and functionality of Network Devices and Network commands
CO 2	Apply subnetting techniques to design various topologies in computer networking
CO 3	Compare various routing algorithms and it's simulation.
CO 4	Understanding the implementation of VLAN's and DHCP servers.
CO 5	Applying Network Address Translation
CO 6	Understand how to implement client-server applications using JAVA.

Assessment Method

Assessment Tool	Experiments	Report/Viva-Voce/ Quiz/MCQ/Lab project	Total
Weightage (%)	25%	15%	40%
End Semester Examination weightage (%)			60%

****

Course Code	Course Name	Course Category	L-T-P	Credits
20CS3183	**Software Engineering Laboratory**	**PCC**	**0-0-3**	**1.5**

Course Learning Objectives:

1. To get familiar with objective and abstract of the project.
2. To get familiar with preparing a document which is used before starting the project
3. Prepare System Design & Architecture
4. Identify & draw different UML based diagrams
5. To understand actual system using analysis model
6. To understand the project implementation according to MVC architecture
7. To understand various testing techniques.

List of Experiments:

1. Identify the Objective and Abstract of the project based on the given Scenarios
2. Prepare Software Requirement Analysis document to respective project
3. Compute function point of the selected project
4. Prepare System Design & Architecture design

5. UML based design diagrams
6. Prepare Data Flow Diagrams of the selected project
7. Complete design part of the selected project
8. Design of the Test Cases of the selected project
9. On completion of coding prepare test cases and perform black-box testing , record the defects and its time of identification
10. Do vulnerability analysis of the selected project

Course Outcomes

At the end of the course, the student will be able to

CO 1	Objective and Abstract of the project
CO 2	SRS document which contains detailed information about the given project
CO 3	Design document which contains detailed diagrams of System design and System Architecture of the given project
CO 4	Design document which contains detailed diagrams of Use case, Class, Sequence, Activity, State, State transition Diagrams
CO 5	E-R diagram, data-flow diagram, state-transition diagram for the project
CO 6	Understanding about Unit testing, regression testing, integration testing, validation and system testing
CO 7	Implementing the given project with Advanced Java programming according to MVC Architecture

Assessment Method

Assessment Tool	Experiments	Report/Viva-Voce/ Quiz/MCQ/Lab project	Total
Weightage (%)	25%	15%	40%
End Semester Examination weightage (%)			60%

****

Year & Sem:	Course Code:	Course Name:	L – T -P:	Credits
E3 & SEM1	20EG3182	ENGLISH LANGUAGE COMMUNICATION SKILLS LAB-II	0 – 0 – 3	1.5

Course objectives:

1. To improve group discussion skills of the students
2. To help the students to write their CV and Internship application
3. To improve the telephonic etiquettes of the students
4. To help the students to take decision on their career

COURSE CONTENT

UNIT-I: **(06 Contact Hours)**

Group Discussion - How to think and analyze - How to initiate a topic - How to continue a topic - How to support or reject a point-of-view - How to defend your position - Managing distractions and mediating between contenders - How to summarize & conclude

UNIT-II: **(06 Contact Hours)**

Telephonic conversation & Etiquettes - How to introduce oneself - How to introduce the main issue - How to keep the other person engaged - How to convince the other person - How to complain without irritating. - Giving assurance and asking for clarification - How to end a formal telephonic conversation

UNIT-III: **(06 Contact Hours)**

Career Planning & Job-Skill Analysis - ASK: Talking about one's Attitudes, Knowledge, & Skills - SMART goals - Reading & Analysis of Job Advertisements

UNIT-IV: **(06 Contact Hours)**

CV & Resume Writing - Difference between CV & Resume - Writing CV - Writing Resume - Writing Cover Letter

UNIT-V: **(06 Contact Hours)**

Application for Internship - Application for internship in Academic Labs - Application for internship in Industries - Follow up the Application with reminders and requests

UNIT-VI: **(06 Contact Hours)**

Interview Skills - Preparation for the Interview - Frequently asked questions - Dress Codes, Appearance, and Etiquettes. 6.4 Facing the Interview

References:

1. *Business Communication Today*, 12th Edition, Courtland L Bovee & John Thill, Pearson
2. British Council Material on Career Planning & Interviews
3. *Master the Group Discussion & Personal Interview - Complete Discussion on the topics asked by reputed B-schools & IIMs* by Sheetal Desarda, Notion Press
4. *Group Discussion and Interview Skills* by Priyadarshi Patnaik , Cambridge University Press India
5. *The Ultimate Guide to Internships: 100 Steps to Get a Great Internship and Thrive in It* by Eric Woodard
6. Telephone Etiquette by Robert DeGroot

Course outcomes: At the end of the course, the student will be able to

CO 1	Get used to a variety of GDs to understand the principles, finer nuances, and intricacies of the art
CO 2	Get exhaustive information on how to prepare for internship and interview
CO 3	Write his/her CV to remain well-prepared for the interviews

CO 4	Take decision on his/her career goals and plans
CO 5	Attain professional speaking skills to enhance his/her employability skills.

Assessment Method:

Course Nature: LABORATORY

Internal Assessment (40 Marks)	**External Assessment (**60 Marks)
Record Writing – 10 Marks	Reading Comprehension – 15 Marks
Attendance – 10 Marks	Writing – 30 Marks
Continuous Assessment (Listening – 10 Marks + Oral Presentations – 10 Marks)	Speaking (Viva-Voce) – 15 Marks

****

ENGINEERING THIRD YEAR: SEMESTER-II

ourse code	**Course name**	**Course Category**	**L-T-P**	**Credits**
20CS3201	**Cryptography and Network Security**	**PCC**	**3-1-0**	**4**

Course Learning Objectives:

1. To understand basics of Cryptography and Network Security.
2. To be able to secure a message over insecure channel by various means.
3. To learn about how to maintain the Confidentiality, Integrity and Availability of a data.
4. To understand various protocols for network security to protect against the threats in the networks.

Course Content:

Unit 1: (10 Contact hours)

Introduction to security attacks, services and mechanism, introduction to cryptography - Conventional Encryption: Conventional encryption model, classical encryption techniques - substitution ciphers and transposition ciphers, cryptanalysis, stream and block ciphers ,Modern Block Ciphers: Block ciphers principals, Shannon's theory of confusion and diffusion, fiestal structure, data encryption standard(DES), strength of DES, differential and linear crypt analysis of DES, block cipher modes of operations, triple DES, AES.

Unit II: (8 Contact hours)

Confidentiality using conventional encryption, traffic confidentiality, key distribution, random number generation, Introduction to graph, ring and field, prime and relative prime numbers, modular arithmetic, Fermat's and Euler's theorem, primality testing, Euclid's Algorithm, Chinese Remainder theorem, discrete algorithms.

Unit III: **(7 Contact hours)**

Principles of public key crypto systems, RSA algorithm, security of RSA, key management, Diffle-Hellman key exchange algorithm, introductory idea of Elliptic curve cryptography, Elgamel encryption, Message Authentication and Hash Function: Authentication requirements, authentication functions, message authentication code, hash functions, birthday attacks, security of hash functions and MACS.

Unit IV: **(7 Contact hours)**

MD5 message digest algorithm, Secure hash algorithm (SHA), Digital Signatures: Digital Signatures - authentication protocols - digital signature standards (DSS) - proof of digital signature algorithm, Authentication Applications: Kerberos and X.509, directory authentication service, electronic mail security, pretty good privacy (PGP), S/MIME.

Unit V: **(7 Contact hours)**

IP Security: Architecture, Authentication header, Encapsulating security payloads, combining security associations, key management.

Unit VI: **(7 Contact hours)**

Web Security: Secure socket layer and transport layer security, secure electronic transaction (SET), System Security: Intruders - Viruses and related threats, IDS.

Learning Resources

Text books:

1. William Stallings, "Crpyptography and Network security Principles and Practices", Pearson/PHI.
2. Wade Trappe, Lawrence C Washington, " Introduction to Cryptography with coding theory", Pearson**.**

Reference Books:

6. W. Mao, "Modern Cryptography – Theory and Practice", Pearson Education.
7. Charles P. Pfleeger, Shari Lawrence Pfleeger – Security in computing – Prentice Hall of India.

Web resources:

1. http://nptel.ac.in/courses/106105031/ lecture by Dr. Debdeep Mukhopadhyay IIT Kharagpur

2. https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-033-computer-system-engineering-spring-2009/video-lectures/ lecture by Prof. Robert Morris and Prof. Samuel Madden MIT.

Course Outcomes: At the end of the course, the student will be able to

CO 1	Implement security of the data over the network.
CO 2	Explore emerging areas of cryptography and network security.
CO 3	Implement various networking protocols.
CO 4	Demonstrate how to protect any network from the threats in the world.

For Theory courses only:

Course Nature	**Theory**
Assessment Method	

Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

****

Course code	Course name	Course Category	L-T-P	Credits
20CS3202	**Artificial Intelligence**	**PEC**	**3-1-0**	**4**

Course Learning Objectives:

1. Definitions of Artificial Intelligence, Different Perspectives, Historical background
2. To understand those elements constituting problems and learn to solve it by various searching techniques
3. To understand those formal methods for representing the knowledge and the process of inference to derive new representations of the knowledge to deduce what to do
4. To understand the notion of planning in AI and some techniques in the classical planning system
5. To understand the notion of uncertainty and some of probabilistic reasoning methods to deduce inferences under uncertainty
6. To understand some of those mechanisms by which an AI system can improve it's behavior through its experience

UNIT I:

Introduction to AI Problems: AI technique, Criteria for success. Problems; Problem Space and Search: Defining the problem as a state space search, Production as a systems, Problem characteristics, Production system characteristics (6 Hours)

Unit II: (6 Hours)

Heuristic Search Techniques: Generate and Test, Hill climbing, Best first search, Problem reduction, Constraint satisfaction, Means ends analysis.

UNIT III **(12 Hours)**

Knowledge Representation: Representation and mappings, Approaches to knowledge representation; Issues in knowledge representation. Using Predicate Logic: Representing simple facts in logic, Representing instance and IS-A relationships, Computable functions and predicates, Resolution, Natural deduction, Forward vs. Backward reasoning.

UNIT IV: (6 Hours)

Different Knowledge Representation Schemes: Semantic nets, Frames, Conceptual dependency, Scripts

UNIT V: (9 Hours)

Natural Language Processing: Overview of linguistics, Grammars and languages, Basic parsing techniques, Transitional networks, Semantic analysis and representation structures, Brief introduction on discourse and pragmatic processing;

UNIT VI: **(6 Hours)**

Expert System Architecture: Characteristic features of expert systems, History, Applications, Rule based system architecture. General Concepts in Knowledge Acquisition: Types of learning, General learning model, Performance measures.

Text Books:

- Elaine Rich, Kevin Knight, "Artificial Intelligence", 3rd Edition, Tata McGraw-Hill, 2009.
- Dan W. Patterson, "Introduction to Artificial Intelligence & Expert Systems", PHI, 1990.

References:

1. Stuart Russel and Peter Norvid, " Artificial Intelligence : A Modern Approach", Pearson Education, 3rd Edition

Web References

1. Berkley University, "Artificial Intelligence", https://courses.edx.org/courses/BerkeleyX/CS188.1x-4/1T2015/course/
2. MIT, "Artificial Intelligence", FALL 2010 https://www.youtube.com/playlist?list=PLUl4u3cNGP63gFHB6xb-kVBiQHYe_4hSi
3. "Introduction to Artificial Intelligence", https://classroom.udacity.com/courses/cs271

Course Outcomes: At the end of this course, the student should be able to

CO 1	Identify problems that are amendable to solution by AI methods
CO 2	Identify appropriate AI methods to solve a given problem
CO 3	Formalise a given problem in the language / framework of different AI methods
CO 4	Implement basic AI algorithms
CO 5	Design and carry out an empirical evaluation of different algorithms on a problem formalization, and state the conclusions that the evaluation supports.

Course Nature	**Theory**			
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

****

Year & Sem: **E3 & SEM2**	**Course Code:** **EG3283**	**Course Name:** **ENGLISH LANGUAGE COMMUNICATION SKILLS LAB-III**	**L – T -P:** **0 – 0 – 3**	**Credits** **1.5**

Course objectives:

1. To improve interpersonal skills of the students
2. To help the students to write professional letters and reports
3. To practice the etiquettes to be used at workplace
4. To reward hands on experience on managing meetings
5. To imbibe leadership qualities in the students

COURSE CONTENT

UNIT-I: **(06 Contact Hours)**

Professional Presentation - Collecting & Reading the materials to be presented - Analyzing the main points - Summarizing & concluding - Developing PPT - Delivery of the Presentation

UNIT-II: **(06 Contact Hours)**

Report Writing & Writing Professional Emails & Applications – Routine Reports – Investigative Reports - Professional Emails - Formal Letters and Applications

UNIT-III: **(06 Contact Hours)**

Agenda, Meetings, & Minutes - Setting the agenda for a meeting - Managing a meeting - Keynote address & vote of thanks - Publishing the minutes

UNIT-IV: **(06 Contact Hours)**

People skills and small talks (2 minutes) - Talking to professional executives - Talking to colleagues - Talking to the boss - Talking to your team - Talking to the media delegates

UNIT-V: **(06 Contact Hours)**

Corporate Etiquettes - How to introduce & greet - How to raise a question - How to clarify a doubt - How to say "yes" or "no" - Rapport building - Dining & winning - Counseling somebody - How to influence & motivate

UNIT-VI: **(06 Contact Hours)**

Life Skills - Leadership communication - Interpersonal communication - Stress management - Time Management

References:

1. *Business Communication Today*, 12th Edition, Courtland L Bovee & John Thill, Pearson
2. British Council Material on communication
3. Training in Interpersonal Skills: Tips f: Tips for Managing People at Work by Robbins and Hunsaker
4. Soft Skills for Everyone, with CD Paperback –by Jeff Butterfield
5. Communication for business by Shirley Taylor, Pearson

Course outcomes: At the end of the course, the student will be able to

CO 1	The art of professional presentation
CO 2	Write professional reports and letters
CO 3	Conduct a formal meeting
CO 4	Develop people skills and corporate etiquettes
CO 5	Gain the basic knowledge about leadership communication, stress management and time management

Assessment Method:

Course Nature: LABORATORY

Internal Assessment (40 Marks)		**External Assessment (**60 Marks)	
Record Writing	– 10 Marks	Reading Comprehension	– 15 Marks
Attendance	– 10 Marks	Writing	– 30 Marks
Continuous Assessment (Listening – 10 Marks + Oral Presentations – 10 Marks)		Speaking (Viva-Voce)	– 15 Marks

****

Course code	**Course name**	**Course Category**	**L-T-P**	**Credits**
20CS3203	**Career Development Course**	**MC**	**2: 0: 0**	**0**

Course Learning Objectives:

1. To enable the students for their competitive exams
2. To enhance their capability in aptitude and reasoning & programming.
3. To develop their reasoning skill.
4. To prepare them for all type of competitive exams

Course Contents:

Unit I: (1.5 hours)

Number system: Base System, Exponents, Factorials, LCM & HCF, Properties of Numbers, Remainders, Successive Divisions

Sequence & Series: Arithmetic Progression, Harmonic Progression, Geometric Progression

Programming in C

Unit II: (8 hours)

Arithmetic: Averages, Clocks & Calendars, Simple Interest & Compoud Interest, Mixture & Alligations, Percentages, Profit, Loss & Discounts, Ratio & Proportion, Speed, Time & Distance, Time & Work

Programming in JAVA

Algebra: Binomial Theorem, Complex Numbers, Functions, Higher Degree Equations, Inequalities , Linear Equations, Logarithm, Quadratic Equations

Programming in Phython

Unit III: (6 hours)

Geometry: Mensuration, Lines & Angles, Circles, Polygons, Triangles, Co-ordinate Geometry, Trigonometry

Probability & Statistics: Mean, Median & Mode, Permutation & Combination, Probability Set Theory & Venn Diagram

Programming in C++

Unit IV: (7 hours)

Logical Reasoning: Logical Sequence, Premise, Assumption & Conclusion, Binary Logic, Blood Relations, Linear & Matrix Arrangement, Seating Arrangement, Coding & Decoding, Statements & Assumptions Puzzles

Analytical Reasoning: Course of Action Fact, Inference & Judgement, Logical Deduction, Statement & Assumption, Strong & Weak Arguments, Syllogism

Unit V: (4.5 hours)

Data Interpretation: Charts (Column, Pie & Bar), Tables Graphs (Line & Area), Venn Diagram, Data Sufficiency. Reading Comprehension

Unit VI: **(3 hours)**

Verbal Ability: Cloze Test Error Spotting, Fill in the blanks, Sentence Correction, Word Usage, Para jumbles, Paragraph Completion, Paragraph Summary

Learning resources

Text book:

1. Sarvesh K Verma, *'Quantitative Aptitude Quantum CAT'*, arihant publications
2. Arun Sharma, Meenakshi Upadhyay, *' Verbal Ability and Reading Comprehension'* , McGraw Hill publications
3. Arun Sharma, *'Data Interpretation'*, McGraw Hill publications
4. Arun Sharma, *'Logical Reasoning'*, McGraw Hill publications

Reference books:

1. Nishit K Sinha, 'Logical Reasoning and Data Interpretation', Pearson publications
2. Arun Sharma, *'Quantitative Aptitude'*, McGraw Hill publications

Web resources:

1. https://unacademy.com/

Course outcomes: At the end of the course, the student will be able to

CO 1	Improve aptitude, problem solving skills and reasoning abilities
CO 2	Improve Verbal ability skills, Data interpretation skills
CO 3	Understand the basic techniques required for solving Reading Comprehension
CO 4	Familiarize with the written tests of competitive exams, campus placements and PSUs
CO 5	Collectively solve problems in teams and group
CO 6	Adopt and acquire new techniques in solving problem

Assessment Method

Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	Nil	Nil	100	100

****

Engineering year Final year semster I

Course Code	Course Name	Course Category	L-T-P	Credits
20CS4101	**MACHINE LEARNING**	PEC	**3-1-0**	**4**

Objectives:

- To understand the basic theory underlying machine learning.
- To be able to formulate machine learning problems corresponding to different applications.
- To understand a range of machine learning algorithms along with their strengths and weaknesses.
- To be able to apply machine learning algorithms to solve problems of moderate complexity.

Course Outcomes:

- Ability to understand what is learning and why it is essential to the design of intelligent machines.
- Ability to design and implement various machine learning algorithms in a wide range of real-world applications.

UNIT I: INTRODUCTION

Learning Problems – Perspectives and Issues - A brief introduction to Machine Learnin`g, Supervised Learning, Unsupervised Learning, Reinforcement Learning – Concept Learning – Version Spaces and Candidate Eliminations – Inductive bias – Decision Tree learning

UNIT II: NEURAL NETWORKS AND GENETIC ALGORITHMS

Neural Network Representation – Problems – Perceptrons – Multilayer Networks and Back Propagation Algorithms – Advanced Topics – Genetic Algorithms – Hypothesis Space Search – Genetic Programming – Models of Evalution and Learning.

UNIT III: BAYESIAN LEARNING

Bayes Theorem – Concept Learning – Maximum Likelihood – Minimum Description Length Principle – Bayes Optimal Classifier – Gibbs Algorithm – Naïve Bayes Classifier – Bayesian Belief Network – EM Algorithm – Probability Learning – Sample Complexity – Finite and Infinite Hypothesis Spaces – Mistake Bound Model.

UNIT IV : COMPUTATIONAL LEARNING

Probability Learning – Sample Complexity – Finite and Infinite Hypothesis Spaces – Mistake Bound Model. Learning Sets of Rules – Sequential Covering Algorithm – Learning Rule Set – First Order Rules – Sets of First Order Rules.

UNIT V: INSTANCE BASED LEARNING

K- Nearest Neighbour Learning – Locally weighted Regression – Radial Bases Functions – Case Based Learning.

UNIT VI: ADVANCED LEARNING

SVM – Formulation, SVM – Interpretation & Analysis, SVMs for Linearly Non-Separable Data, SVM Kernels. Reinforcement Learning – Task – Q-Learning – Temporal Difference Learning

TEXT BOOKS:

1. Machine Learning – Tom M. Mitchell, - MGH
2. Machine Learning: An Algorithmic Perspective, Stephen Marsland, Taylor & Francis

REFERENCE BOOKS

1. Gareth James, Daniela Witten, Trevor Hastie, Robert Tibshirani, Trevor Hastie," *An Introduction to Statistical Learning: with Applications in R",* Springer, First Edition.
2. Kevin Murphy, "*Machine learning: a probabilistic perspective"*, MIT Press, First Edition.
3. Christopher Bishop, "*pattern recognition and machine learning",* Springer, First Edition.

****

LIST OF PROFESSIONAL ELECTIVES COURSE(PEC)
Elective group-1

Course Code	Course Name	Course Category	L-T-P	Credits
20CS3121	Data Mining	PEC	3-0-0	3

Course Learning Objectives

4. Interpret the contribution of data warehousing and data mining to the decision-support level of organizations
5. Categorize and carefully differentiate between situations for applying different data-mining techniques: frequent pattern mining, association, correlation, classification, prediction, and cluster and outlier analysis
6. Design and implement systems for data mining
7. Evaluate the performance of different data-mining algorithms
8. Propose data-mining solutions for different applications

Course Content:

Unit - I: (6 Contact hours)

Data Mining: Data mining functionalities, Integration of a data mining system with a database or data warehouse systems, Classification of data mining systems, Data mining task primitives, Major issues in data mining. Data Processing: Data cleaning, Data integration and transformation, Data reduction, Discretization and concept hierarchy generation.

Unit-II: (9 Contact hours)

Data Warehouse and OLAP Technology: Differences between operational database systems and data warehouses, A Multidimensional data model, Data warehouse architecture, Data warehouse implementation - Efficient computation of data cubes, From Data warehousing to data mining.

Unit - III: (9 Contact hours)

Association Rules in Large Databases: Basic concepts of association rule mining, Efficient and scalable frequent itemset mining methods. Mining Multilevel Association Rules: Mining multidimensional association rules from relational databases and data warehouses, From Association mining to correlation analysis, Constraint based association mining.

Unit – IV: (7 Contact hours)

Classification: Issues regarding classification and prediction, Classification by decision tree induction, Bayesian classification, Rule-Based classification. Prediction: Linear regression, Nonlinear regression, Other regression based methods.

Unit – V: (9 Contact hours)

Cluster Analysis: Basic of cluster analysis, Types of data in cluster analysis, A categorization of major clustering methods, Partitioning methods – k-Means and k-Medoids, Hierarchical method - Agglomerative vs. divisive hierarchical clustering, Distance measures in algorithmic methods, BIRCH, Chameleon, Density based clustering: DBSCAN,

Unit – VI: (5 Contact hours)

Outlier analysis :

Introduction to outlier detection, design and implementation of naïve methods related to data mining techniques.- Statistical distribution based outlier detection, Distance based outlier detection.

Learning resources:

Text book:

- Jiawei Han, MichelineKamber,*"Data Mining: Concepts and Techniques",* Morgan Kaufmann Publishers editor, 2006.
- Pang-Ning Tan, Michael Steinbach, and Vipin Kumar. *Introduction to Data Mining.*
- David J. Hand, HeikkiMannila, PadhraicSmyth ,*"Principles of Data mining"* MIT Press
- Parteek Bhatia "Data Mining and Data Warehousing " 1/e, Cambridge,19

Reference Books:

1. *"Advances in Knowledge Discovery and Data Mining".* AAAI/MIT Press, 1996. Selected papers from conferences and journals, conference tutorials.
2. *Jiawei Han, Micheline Kamber, and Jian Pei.* Data Mining:*Concepts and Techniques* (3rd ed.). Morgan Kaufmann, 2012. eText ISBN: 9780123814807.

Web resources:

http://nptel.ac.in/courses/106106093/35

Course outcomes: At the end of the course, the student will be able to

CO 1	Discover interesting pattern from large amount of data to analyze and extract patterns to solve problems make predictions of outcomes.
CO 2	Evaluate systematically supervised models and algorithms w.r.t their accuracy.
CO 3	Evaluate and implement a wide range of emerging and newly-adopted methodology and technologies to facilitate the knowledge discovery.
CO 4	Design and implement of a data mining applications using sample, realistic data sets and modern tools.
CO 5	Evaluate and select appropriate data mining algorithms apply, interpret and report the output appropriately.

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

****

Course code	**Course name**	**Course Category**	**L-T-P**	**Credi**
20CS3122	**MOBILEAPPLICATION DEVELOPMENT**	**ESC**	**3-0-0**	**3**

Syllabus:

Course Objective:

- Describe those aspects of mobile programming that make it unique from programming for other platforms,

- Critique mobile applications on their design pros and cons,
- Utilize rapid prototyping techniques to design and develop sophisticated mobile interfaces,
- Program mobile applications for the Android operating system that use basic and advanced phone features, and
- Deploy applications to the Android marketplace for distribution.

UNIT I: **(6 Hours)**

J2ME Overview: Java 2 Micro Edition and the World of Java, Introduction to Mobile app Development, Reasons to Build Mobile App, Mobile Application Development Today, Myths of Mobile Application Design. Diving into Mobile: App or Website? Mobile Web Presence

UNIT II : **(12 Hours)**

Android Platform and Development Environment: Intro to Android, Create your First Android App, Layouts, Views and Resources. Text and Scrolling Views. Activities and Intents. Debugging and Testing your App

UNIT III : **(9 Hours)**

User Input Controls, Menus, Screen Navigation, Recycler View, Drawables, Themes and Styles. Material Design

UNIT IV: **(10 Hours)**

Async Task and Async TaskLoaders, Connecting to the Internet, Broadcast Receivers, Services, Notifications, Alarm Managers, Transferring Data Efficiently. Shared Preferences

UNIT V: **(5 Hours)**

SQLite Primer, Store Data using SQLite, Content Providers, Using Loaders to Load and Display Data, Permissions, Performance and Security.

UNIT VI : **(10 Hours)**

Kotlin : Introduction to kotlin, Environment Setup , Architecture, Functions, Classes, Visibility Control, Inheritance and Interfaces, Extensions, Data and Sealed Classes, Delegation, Example to build basic apps using Kotlin.

Text Books

1. Professional Mobile Application Development by Jeff McWherter, Scott Gowell Wiley india pvt.ltd 2013
2. https://google-developer-training.gitbooks.io/android-developer-fundamentals-course-practicals/content/en/
3. https://developers.google.com/training/courses/android-fundamentals
4. https://kotlinlang.org/docs/tutorials/edu-tools-learner.html

Open Source Videos Link:

Course Name: Android Fundamentals Developer - Videos

https://www.youtube.com/playlist?list=PLlyCyjh2pUe9wv-hU4my-Nen_SvXIzxGB

Reading Material Open Source Link:

https://google-developer-training.gitbooks.io/android-developer-fundamentals-course-concepts/content/en/

https://www.tutorialspoint.com/kotlin

Tutorial Hours (Practical Problem sets) Open Source Link:

https://google-developer-training.gitbooks.io/android-developer-fundamentals-course-practicals/content/en/

Course outcomes: At the end of the course, the student will be able to

CO 1	Understand importance of mobile presence
CO 2	Will be able to design mobile applications
CO 3	Understand the Android Studio installation, Project structure, creating activities, testing apps and using the Android Support Library
CO 4	Understand that how to get input from the user, implement navigation strategies, use themes and styles, test your user interface, and follow Material Design principles
CO 5	Understand how to do background work, how to schedule tasks, and how to trigger events
CO 6	Understand the difference between other languages and kotlin, how to build apps using kotlin.

For Theory courses only:

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

****

Course Code	Course Name	Course Category	L-T-P	Credits
20CS3123	**Distributed Computing**	**PEC**	**3-0-0**	**3**

Course Learning Objectives:

1. To expose students to both the abstraction and details of file systems
2. To introduce concepts related to distributed computing systems.
3. To focus on performance and flexibility issues related to systems design decisions.
4. To prepare students for life-long learning

5. To evaluate how and not just the memorize the details.
6. To expose students to current literature in distributed systems.

Course Content:

UNIT – I **(8 Contact hours)**

Fundamentals: Evolution of Distributed Computing Systems, System models, issues in design of Distributed Systems, Distributed computing environment, web based distributed model, computer networks related to distributed systems and web based protocols

UNIT-II **(8 Contact hours)**

Message Passing: Inter process Communication, Desirable Features of Good Message-Passing Systems, Issues in IPC by Message, Synchronization, Buffering, Multidatagram Messages, Encoding and Decoding of Message Data, Process Addressing, Failure Handling, Group Communication

UNIT-III **(7 Contact hours)**

Remote Procedure Calls: The RPC Model, Transparency of RPC, Implementing RPC Mechanism, Stub Generation, RPC Messages, Server Management, Communication Protocols for RPCs, Complicated RPCs, Client-Server Binding, Exception Handling,Some Special Types of RPCs, Lightweight RPC, Optimization for Better Performance.

UNIT- IV **(8 Contact hours)**

Distributed Shared Memory: Design and Implementation issues of DSM, Granularity, Structure of Shared memory Space, Consistency Models, replacement Strategy, Thrashing, Other Approaches to DSM, Advantages of DSM.

UNIT – V **(8 Contact hours)**

Synchronization: Clock Synchronization, Event Ordering, Mutual Exclusion, Election Algorithms. Constant.

Resource and Process Management: Desirable Features of a good global scheduling algorithm, Load Balancing approach, Load Sharing Approach, Process Migration, Threads, Processor allocation, Real time distributed Systems.

UNIT – VI **(8 Contact hours)**

Distributed File Systems: Desirable Features of a good Distributed File Systems, File-sharing Semantics, File Models, File Accessing Models,ile caching Schemes, File Replication, Design Principles, Sun's network file system, Andrews file system, comparison of NFS and AFS.

Learning resources

Text book:

1. Ajay D. Kshemkalyani, Mukesh Singhal , "Distributed Computing: Principles, Algorithms, and Systems Reissue Edition",

2. Sukumar Ghosh, Distributed Systems: An Algorithmic Approach, Second Edition

Reference Books:

1. Distributed OS by Pradeep K. Sinha (PHI)
2. Tanenbaum S.: Distributed Operating Systems, Pearson Education
3. Tanenbaum S. Maarten V.S.: Distributed Systems Principles and Paradigms, (Pearson Education)
4. George Coulouris, Jean Dollimore. Tim Kindberg: Distributed Systems concepts and design.

Web resources:

https://eclass.uoa.gr/modules/document/file.php/D245/2015/DistrComp.pdf

Course outcomes: At the end of the course, the student will be able to

CO 1	Study software components of distributed computing systems.
CO 2	Know about the communication and interconnection architecture of multiple computer systems.
CO 3	Recognize the inherent difficulties that arise due to distributed-ness of computing resources.

CO 4	Understanding of networks & protocols, mobile & wireless computing and their applications to real world problems
CO 5	Able to use pointers in C programming

For Theory courses only:

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

****

Course Code	**Course Name**	**Course Category**	**L-T-P**	**Credits**
20CS3124	**Advanced Computer Architecture**	PEC	**3-0-0**	**3**

Course Learning Objectives:

1. To understand the Concept of Parallel Processing and its applications.
2. To understand the micro-architectural design of processors.
3. To learn about the various techniques used to obtain performance improvement and power savings in current processors.
4. To develop the Pipelining Concept for a given set of Instructions.
5. To distinguish the performance of pipelining and non pipelining environment in a processor.
6. To ability to improve the performance of applications on modern and high performance computers.

Course Content:

UNIT – I **(10 Contact hours)**

Introduction and Pipelinig: Defining computer architecture; Trends in Technology, power in Integrated Circuits and cost; Dependability; Measuring, reporting and summarizing Performance; Quantitative Principles of computer design;Pipeline hazards; Implementation of pipeline; What makes pipelining hard to implement?

UNIT-II **(10 Contact hours)**

Instruction –Level Parallelism: ILP: Concepts and challenges; Basic Compiler Techniques for exposing ILP; Reducing Branch costs with prediction; Overcoming Data hazards with Dynamic scheduling; Hardware-based speculation

UNIT-III **(8 Contact hours)**

Instruction –Level Parallelism – 2: Exploiting ILP using multiple issue and static scheduling; Exploiting ILP using dynamic scheduling, multiple issue and speculation; Advanced Techniques for instruction delivery and Speculation; The Intel Pentium 4 as example.

UNIT – IV **(12 Contact hours)**

Multiprocessors and Thread –Level Parallelism: Introduction; Symmetric shared-memory architectures; Performance of symmetric shared–memory multiprocessors; Distributed shared memory and directory-based coherence; Basics of synchronization; Models of Memory Consistency.

UNIT – V **(12 Contact hours) Memory**

Hierarchy and Design : Introduction; Cache performance; Cache Optimizations, Virtual

memory;Advanced optimizations of Cache performance; Memory technology and optimizations; Protection: Virtual memory and virtual machines.

UNIT – VI **(10 Contact hours)**

Hardware and Software for VLIW and EPIC: Introduction: Exploiting Instruction-Level Parallelism Statically; Detecting and Enhancing Loop-Level Parallelism; Scheduling and Structuring Code for Parallelism; Hardware Support for Exposing Parallelism: Predicated Instructions; Hardware Support for Compiler Speculation; The Intel IA-64 Architecture and Itanium Processor;

Learning resources

Text book:

1. John L Hennessey and David A Patterson, "Computer Architecture A Quantitative Approach", Morgan Kaufmann/ Elsevier, Fifth Edition, 2012

Reference Books:

1. Kai Hwang and Faye Briggs, "Computer Architecture and Parallel Processing", Mc Graw-Hill International Edition, 2000.
2. Sima D, Fountain T and Kacsuk P, "Advanced Computer Architectures: A Design Space Approach", Addison Wesley, 2000.

Web resources:

Higher Performance Computer Architecture, NPTEL Course
https://nptel.ac.in/courses/106105033/1

Course outcomes: At the end of the course, the student will be able to

CO 1	Evaluate performance of different architectures with respect to various parameters
CO 2	Analyze performance of different ILP techniques.
CO 3	Identify cache and memory related issues in multi-processors
CO 4	Design the overall organization of cache and virtual memories, and pipelined processors.
CO 5	Describe the challenges faced in the implementation of these high performance system
CO 6	Improve application performance for different cpu architectures

For Theory courses only:

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

LIST OF PROFESSIONAL ELECTIVES COURSE(PEC)

Elective group-2

Course Code	Course Name	Course Category	L-T-P	Credits
20CS3221	OBJECT ORIENTED ANALYSIS & DESIGN(OOAD)	PEC	3-0-0	**3**

Course Learning Objectives:

1. Master the implementation of different Models

2. Be familiar with models,relationships,roles,types and interfaces
3. Demonstrate understanding the abstractions of various system models
4. Demonstrate understanding of various common modeling techniques
5. Implement various object models designing in more than one manner.
6. Choose the appropriate model for a specified application and different mechanisms

Course Content:

UNIT I : Introduction to UML **(6 Contact hours)**
Importance of modeling, principles of modeling, object oriented modeling, conceptual model of the UML, Architecture, and Software Development Life Cycle.
UNIT II : Basic Structural Modeling **(9 Contact hours)**
Classes Relationships, Common Mechanisms, and diagrams.
Advanced Structural Modeling: Advanced classes, advanced relationships, Interfaces, Types and Roles, Packages.
UNIT III: Class & Object Diagrams **(7 Contact hours)**
Terms, concepts, modeling techniques for class & object diagrams
UNIT IV : Basic Behavioral Modeling **(7 Contact hours)**
Interactions, Interaction diagrams, Use cases, Use case diagrams, Activity Diagrams.
UNIT V : Advanced Behavioral Modeling **(8 Contact hours)**
Events and signals, state machines, processes and threads, time and space, state chart diagrams**.**
UNIT VI :Architectural Modeling **(8 Contact hours)**
Component, Deployment, Component diagrams and Deployment diagrams and case study.

Learning resources:
Text Books:

1. Grady Booch, James Rumbaugh, Ivar Jacobson: *'The Unified Modeling Language User Guide'*, Pearson Education.

Refernce books:

1. Meilir Page-Jones, *'Fundamentals of Object Oriented Design in UML'*, Pearson Education.
2. AtulKahate, *'Object Oriented Analysis & Design'*, The McGraw-Hill Companies

Web resources:

1. IIT Kharagpur, July 22 2016, *'Foundations of the Object Model'*,
   a. URL:**https://nptel.ac.in/courses/106105153/10**
2. IIT Kharagpur, August 12 2016, *'Overview of UML'*,
   a. URL: **https://www.youtube.com/watch?time_continue=31&v=iN4Ft6loL7o**
3. IIT Kharagpur, July 29 2016, *'Relationship among objects'*,
   a. URL: **https://nptel.ac.in/courses/106105153/20**

Course outcomes: At the end of the course, the student will be able to

CO 1	Define basic terms necessery for modeling computer systems
CO 2	Collect requirements and prepare their scenarios
CO 3	Draw diagrams by UML
CO 4	Prepare and use of design patterns
CO 5	Prepare supporting documentation.
CO 6	Create a New Models as per requirements.

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

****

Course Code	Course Name	Course Category	L-T-P	Credits
20CS3222	DISTRIBUTED SYSTEMS	PEC	3-0-0	3

Course Learning Objectives:

1. To learn the fundamentals of Distributed Systems
2. To learn distributed services such as the world-wide web
3. To learn examples of research and commercial distributed systems
4. To learn about distributed algorithms and distributed file systems and distributed
5. databases, security and protection
6. To expose students to past and current research issues in the field of distributed systems
7. challenges in cloud computing.
8. To create an awareness of the fundamental technical challenges in advanced distributed
9. systems design and implementation.

Course Contents:

UNIT I **(7 Contact hours)**

Definition of distributed systems, Goals of distributed system, types of distributed systems, Architectural Styles , types of architectures

UNIT II **(8 Contact hours)**

Threads in Distributed Systems ,Virtualization, clients , servers , client server design issues, code migration.

UNIT III **(8 Contact hours)**

Distributed systems communication fundamentals, RPC,Message-Oriented Communication Stream-Oriented Communication , Multicast communication .

UNIT IV **(9 Contact hours)**

Names ,Distributed Hash Tables ,Attribute-based Naming Hierarchical Implementations ,Decentralized Implementations .clock synchronization,logical clocks, mutual exclusion, election algorithms, atomic transaction, deadlocks in distributed systems.

UNIT V **(8 Contact hours)**

Distributed file systems, distributed file system design, distributed file system implementation, trends in distributed file system.

UNIT VI **(9 Contact hours)**

Distributed shared memory, introduction, what is shared memory, consistency models, page based distributed shared memory, shared-variable distributed shared memory, object based distributed shared memory.

Text Book:

1. Andrew S. Tanenbaum, "Distributed Operating Systems", PHI, Third Edition.
2. George Coulouris,"Distributed Systems: Concepts and Design"Pearson,5th Edition

References:

1. William Stallings. "Operating Systems, Internal and Design principles", Pearson india,Fourth Edition,

Video References:

1. NPTEL Lecture: Prof. Ananthanarayana V.S https://nptel.ac.in/courses/106106107
2. NPTEL Lecture: Dr.Rajiv Misra https://nptel.ac.in/courses/106104182

Course Outcomes: At the end of the course, the student will be able to

CO 1	Demonstrate knowledge of the basic elements and concepts related to distributed system technologies
CO 2	Demonstrate knowledge of the core architectural aspects of distributed systems
CO 3	Design and implement distributed applications
CO 4	Demonstrate knowledge of details the main underlying components of distributed systems (such as RPC, file systems)
CO 5	Demonstrate knowledge of details the main underlying components of distributed systems (such as RPC, file systems)
CO 6	Demonstrate experience in building large-scale distributed applications.

For Theory courses only:

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

Course Code	**Course Name**	**Course Category**	**L-T-P**	**Credits**
20CS3223	Real Time Operating Systems	PEC	3-0-0	3

Course learning Objectives:

1. To learn the fundamentals of Operating Systems.
2. To learn the mechanisms of OS to handle processes and threads and their communication
3. To learn the mechanisms involved in memory management in contemporary OS
4. To gain knowledge on distributed operating system concepts that includes architecture, Mutual exclusion algorithms, deadlock detection algorithms and agreement protocols

5. To know the components and management aspects of concurrency management, To learn programmatically to implement simple OS mechanisms
6. Syllabus deals with issues in real time operating systems, importance of deadlines and concept of task scheduling

Course Content:

UNIT-I: REVIEW OF OPERATING SYSTEMS (8 Contact Hours)

Overview, OS structures, system calls, process cooperation, process communication, semaphores, conditional critical regions, deadlock, processor management, scheduling algorithms, Queuing system model.

UNIT-II AN INTRODUCTION TO RTOS (8 Contact hours)

Introduction to RTOS, types of RTOS, GPOS vs RTOS, applications of RTOS, Issues in real time computing.

UNIT-III: REAL TIME MODELS AND LANGUAGES (8 Contact hours)

Event Based – Process Based and Graph based Models – Pertinent Models – Real Time Languages – RTOS Tasks – RT scheduling - Interrupt processing – Synchronization – Control Blocks – Memory Requirements.

UNIT-IV: REAL TIME KERNEL (8 Contact hours) Principles – Design issues – Polled Loop Systems – RTOS Porting to a Target – Comparison and study of RTOS VX works and µCOS – Case studies.

UNIT-V: RTOS APPLICATION DOMAINS (9 Contact hours)

RTOS for Image Processing – Embedded RTOS for voice over IP – RTOS for fault Tolerant Applications – RTOS for Control Systems.

UNIT-VI: DISTRIBUTED OPERATING SYSTEMS (8 Contact hours) Distributed operating systems concept, file systems, mode of computation, load balancing, event ordering, synchronization, distributed mutual exclusion, drinking philosophers problem, deadlocks in distributed systems.

Text Books:

1. ***R.Mall***, Real Time Systems: Theory and Practice, Pearson Education, 2007.
2. ***William Stallings***, Operating Systems: Internals and Design Principles, 8th edition Pearson Education Limited, 2014 ISBN: 1292061944, 9781292061948

References:

1. ***Tanenbaum***, "Distributed Operating Systems", Pearson Education.
2. ***Jane Liu***, Real Time Systems, Pearson Education, 2000.
3. ***C.M.Krishna*** and K.G.Shin, Real Time Systems, Tata McGraw Hill, 1997.
4. ***Raymond J.A.Bhur, Donald L.Bailey***, "An Introduction to Real Time Systems", PHI 1999.
5. ***D.M Dhamdhere***: Operating systems - A concept based Approach, 3rd Edition, Tata McGraw- Hill, 2012.
6. ***P.C.P. Bhatt***: Introduction to Operating Systems Concepts and Practice, 3rd Edition, PHI, 2010.

Video Reference:

1. NPTEL Lecture: Prof.Rajib Mall https://nptel.ac.in/courses/106105172
2. NPTEL Lecture: Prof. Chester Rebeiro https://nptel.ac.in/courses/106106144

Course Outcomes: At the end of the course, the student will be able to

CO 1	Summarize the issues in real time computing
CO 2	Understand the concept of the tradeoffs in design and implementation concepts used in the development of Operating Systems
CO 3	Understand the differences between the processes and Threads.
CO 4	Understanding of memory management techniques are used in operating system.
CO 5	Understanding of I/O techniques are used in operating system.

CO 6	Analyze the situation of fault occurrence and will be able to apply solutions accordingly.
CO 7	Solve scheduling problems and can apply them in real time applications in industry.
CO 8	Design an RTOS and will be able to interpret the feasibility of a task set to accomplish or not.

For Theory courses only:

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

Course Code	Course Name	Course Category	L-T-P	Credits
20CS3224	**EMBEDED SYSTEMS**	**PEC**	**3-0-0**	**3**

Course Learning Objectives:

1. This course emphasizes on comprehensive treatment of embedded hardware and real time operating systems along with case studies, in tune with the requirements of Industry.
2. The objective of this course is to enable the students to understand embedded-system programming and apply that knowledge to design and develop embedded solutions.

Course Content:

Unit – I **(8 Contact hours)**

Introduction to Embedded Systems: Definition, Applications of ES, Embedded Hardware Units and Devices, Embedded Software, Design Metrics in ES, Challenges in ES Design.

Unit- II **(7 Contact hours)**

Architecture of 8051: 8051 Micro controller Hardware, Input/output Ports and Circuits, External Memory, Counter and Timers, Serial data Input/output, Interrupts and Programming 8051**.**

UNIT –III **(8 Contact hours)**

ARM- Embedded Processor: History, Architecture, Interrupt vector, Programming the ARM, ARM Assembly language, Instruction set, Conditional Execution, Arithmetic and Logical Compare**.**

UNIT – IV **(8 Contact hours)**

ARM PROGRAMMING: Assembly programming, General structure of assembly language, Writing programs, Branch instructions, Loading constrains, load and store instructions, Readonly and read/write Memory, Multiple Register Load and Store.

UNIT – V **(8 Contact hours)**

REAL TIME OPERATING SYSTEMS: Introduction, Tasks and Task States, Tasks and Data, Reentrancy, Semaphores and Shared Data, Inter Process Communication-Message Queues, Mailboxes and Pipes.

UNIT – VI **(7 Contact hours)**

REAL TIME OPERATING SYSTEMS-I : Timer Functions, Events, Memory Management, Interrupt Routines in an RTOS Environment.

TEXT BOOKS:

1. Raj Kamal, "Embedded Systems", 2nd edition, Tata McGraw Hill, 2009.
2. Lyla B Das, "Embedded Systems an Integrated Approach", 1st edition, Pearson, 2012.
3. David E. Simon, "An Embedded Software Primer", 1st edition, Pearson Education, 2008.

REFERENCE BOOKS:

1. Wayne Wolf, "Computers as Components-principles of Embedded Computer system Design", 1st edition, Elseveir, 2009.
2. Labrosse, "Embedding system building blocks", 2rd edition, CMP Publishers, 2007.
3. Kenneth J. Ayala and Thomson, "The 8051 Microcontroller", 3rd edition, Thompson Delmar, Learning, 2008.
4. Frank Vahid, Tony Givargis and John Wiley, "Embedded System Design, Microcontrollers", 3rd edition, Pearson Education, 2008. 5. Michael J. Pont, "Embedded C", Addison Wesley, 2002

Course outcomes: At the end of the course, the student will be able to

CO 1	Understand the concept of embedded system, microcontroller, different components of microcontroller and their interactions
CO 2	Get familiarized with programming environment to develop embedded solutions
CO 3	Program ARM microcontroller to perform various tasks.
CO 4	Understand the key concepts of embedded systems such as I/O, timers, interrupts and interaction with peripheral devices

For Theory courses only:

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

****

Course Code	**Course Name**	**Course Category**	**L-T-P**	**Credits**
20CS3225	**Digital Image Processing**	**PEC**	**3-0-0**	**3**

Course Objectives:

1. To describe the image fundamentals and mathematical transforms necessary for image processing.
2. To explain the image enhancement techniques
3. To analyze images in the frequency domain using various transforms.
4. To explain image restoration procedures.
5. To describe Image Compression and Segmentation used in digital image processing.
6. To describe image feature extraction methods.

Course Outcomes:

1. Be able to apply, design and implement solutions for digital image processing problems.
2. Be able to discuss the strengths and limitations of DIP applications in solving problems with both professional peers and lay clients.

Course content:

UNIT I (8 Content hours)

Digital image fundamentals – Electromagnetic spectrum and imaging, Image acquisition, image formation. Digitization-sampling and quantization, Resolution-pixel, gray scale, spatial, basic relationship between pixels, Distance measure, Mathematical operations on image, Geometrical and spatial transformation.

UNIT II (8 Content hours)

Intensity transformation and spatial filtering: Image enhancement, log transformation, Gamma transformation, Histogram processing, Histogram matching, Special filtering- spatial correlation and convolution, generating spatial filter mask, mage smoothing, Image sharpening-Laplacian filter, Highboost filter. Edge detection- gradient filter, Morphological image processing-erosion, Dilation, opening and closing operations, Boundary extraction, Hole Filling, Extraction of connected components, Thinning, and thickening.

UNIT III (7 Content hours)

Image Restoration-Noise model, Restoration-Mean filter, Geometric filter, median filter, adaptive filter, band pass filter, Notch filter, least mean square filters. Color fundamental-RGB color model, CMY color model, HSI color model. Converting RGB to HSI and vice-versa.

UNIT IV (7 Content hours)

Filtering in Frequency domain-Preliminary concept: Fourier series, Fourier transform, convolution, Sampling, DFT, Enhancement in frequency domain, low pass filter, high pass filter. Computing IDFT from DFT.

UNIT V (8 Content hours)

Image compression fundamental, coding, temporal and spatial redundancy, Error-free (Lossless) and Lossy compression. Image segmentation, Point-line-edge detection. Image gradients operator, canny edge detection, Edge linking and boundary detection, local processing, thresholding, variable thresholding, Region Growing, Texture Segmentation; Region oriented segmentation.

UNIT VI (8 Content hours)

Feature Extraction: Edges – Canny, Sobel; Line detectors, Corners - Harris, Orientation Histogram, SIFT, SURF, Scale-Space Analysis- Image Pyramids, Haar transform. Decision-theoretic and structure descriptors.

Textbooks:

1. Digital Image processing – R.C. Gonzalez & R.E. Woods, Addison Wesley/ Pearson education, 3rd edition.
2. D. A. Forsyth, J. Ponce, Computer Vision: A Modern Approach, PHI Learning 2009.
3. Milan Soanka, Vaclav Hlavac and Roger Boyle, Digital Image Processing and Computer Vision, Cengage Learning.

REFERENCES:

Fundamentals of Digital Image processing – A.K.Jain , PHI

For Theory courses only:

Course Nature	**Theory**

Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

****

LIST OF PROFESSIONAL ELECTIVES COURSE(PEC)

Elective group -3

Course Code	Couse Name	Course Category	L-T-P	Credits
20CS3231	**Information Retrieval**	**PEC**	**3-0-0**	**3**

Course Learning Objectives:

1 To use different information retrieval techniques in various application areas

2 To apply IR principles to locate relevant information large collections of data

3 To analyze performance of retrieval systems when dealing with unmanaged data sources

4 To implement retrieval systems for web search tasks.

Course Content:

Unit I (7Contact hours)

Boolean retrieval. The term vocabulary and postings lists. Dictionaries and tolerant retrieval. Index construction.

Unit II (8 Contact hours)

Index Compression, Scoring, term weighting and the vector space model. Computing scores in a complete search system, Relevance feedback and query expansion.

Unit III (9 Contact hours)

XML retrieval. Probabilistic information retrieval, Text classification. Vector space classification.

Unit IV (7 Contact hours)

Flat clustering, Hierarchical clustering, Matrix decompositions and latent semantic indexing.

Unit V (7 Contact hours)

Web search basics. Web crawling and indexes, Link analysis.

Unit VI (7 Contact hours)

Learning to Rank, Future of web search, Recommender Systems, Content Based Filtering, Collaborative Filtering.

Learning resources:

Text Book:

1.Introduction to Information Retrieval , Christopher D. Manning and Prabhakar Raghavan and Hinrich Schütze, Cambridge University Press, 2008**.**

Reference Books:

1. Information Storage and Retrieval Systems: Theory and Implementation, Kowalski, Gerald, Mark T Maybury, Springer.
2. Modern Information Retrieval, Ricardo Baeza-Yates, Pearson Education, 2007.
3. Information Retrieval: Algorithms and Heuristics, David A Grossman and Ophir Frieder, 2nd Edition, Springer, 2004.

4. Information Retrieval Data Structures and Algorithms, William B Frakes, Ricardo BaezaYates, Pearson Education, 1992. 5. Information Storage & Retieval, Robert Korfhage, John Wiley & Sons
5. C.D. Manning, P. Raghavan, H. Schütze. Introduction to Information Retrieval, Cambridge UP, 2008. (available in the Web, http://nlp.stanford.edu/IR-book/).
6. Kowalski, Gerald, Mark T Maybury: Information Retrieval Systems: Theory and Implementation, Kluwer Academic Press, 1997.

Web Resources:

https://youtu.be/q0srNT_XM_Y?list=PL0ZVw5- GryEkGAQT7lX7oIHqyDPeUyOMQ
https://youtu.be/j789k96g5aQ?list=PL0ZVw5-GryEkGAQT7lX7oIHqyDPeUyOMQ
https://youtu.be/Bl_tfdy6814

Course outcomes: At the end of the course, the student will be able to

CO 1	Ability to identify Data Base Management systems and data ware houses
CO 2	Ability to use knowledge of data structures and indexing methods in information retrieval Systems
CO 3	Ability to choose clustering and searching techniques for different data base systems
CO 4	Ability to Explain different types of search algorithms like Hardware text search systems and software text search systems
CO 5	understand the underlined problems related to IR

For Theory courses only:

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly Tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

****

Course Code	**Course Name**	**Course Category**	**L-T-P**	**Credits**
20CS3232	**Software Testing**	**PEC**	**3-0-0**	**3**

Course Learning Objectives:

1. To understand the basic concepts of software engineering, life cycle models
2. To understand in detail about the requirement analysis and requirement engineering processes and testing methods.
3. To understand the concepts and principles involved in software design .
4. To understand the concepts and various types of software testing and project implementation techniques.
5. To understand the techniques involved in software project management and Risk management.
6. To understand the scope of test automation and tools for testing.

Course Content:

UNIT – I **(8 Contact hours)**

Introduction to software testing: testing as an engineering activity, testing as a process, testing axioms, basic definitions, software testing principles, tester's role in a software development organization.

UNIT-II **(8 Contact hours)**

Introduction to software defects: origin of defects, costs of defects, defect classes, defect repository and test design, defect examples, developer/tester support for developing a defect repository, defect prevention techniques.

UNIT-III **(12 Contact hours)**

test case design : test case design strategies: black box approach to test design, random testing, requirements based testing, boundary value analysis, equivalence class partitioning, state-based testing, cause-effect graph, compatibility testing, user documentation testing, domain testing, using white box approach to test design: test adequacy criteria, static testing vs. structural testing, code functional testing, coverage and control-flow graph, covering-code logic, paths, code-complexity testing, evaluating test-adequacy criteria.

UNIT – IV **(12 Contact hours)**

Levels of testing: need of levels of testing, unit testing,designing the unit tests, the test harness, running the unit tests and recording results, integration tests, designing integration tests, integration test planning,defect bash elimination system testing, acceptance testing, performance testing, regression testing, ad-hoc testing, alpha-beta tests, testing Object Oriented systems-usability and accessibility testing, configuration testing, compatibility testing, testing the documentation, website testing.

UNIT – V **(10 Contact hours)**

Test management: People and organizational issues in testing, organization structure for testing teams, testing services,test plan components, test plan attachments, locating test items, test management, test process, reporting test results, the role of three groups in test planning and policy development-introducing test specialist, skills needed by test specialist, building a testing group.

UNIT – VI **(10 Contact hours)**

Test Automation: software test automation, skills needed for automation, scope of automation, design and architecture for automation, requirements for a test tool, challenges in automation, test metrics and measurements, project progress and productivity metrics.

Learning resources:

Text books:

1. SrinivasanDesikan and gopalaswamyramesh, "Software testing-principles and practices", Pearson Education, 2006
2. Ron Patton, "Software testing", Second Edition, Sams Publishing, Pearson education, 2007

References:

1. Ilene Burnstein, "Practical software testing", Springer International Edition, 2003
2. Edward Kit, "Software testing in real world- improving the process", Pearson Education, 1995
3. Boris Beizer, "Software testing techniques", Second Edition,
4. AdityaP.Mathur, "Foundations of software testing-fundamental algorithms and techniques", Pearson Education, 2008

Web resources:

1. https://nptel.ac.in/courses/106105171/

Course outcomes: At the end of the course, the student will be able to

CO 1	Various test procesess and continuous quality improvement
CO 2	Methods of test generation from requirements

CO 3	Behavior modeling using UML
CO 4	Test adequacy assessment using: control flow, data flow, and program mutations
CO 5	The use of various test tools
CO 6	Application of software testing techniques in commercial environments

For Theory courses only:

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

Course Code	Course Name	Course Category	L-T-P	Credits
20CS3233	**Mobile Computing**	**PEC**	**3-0-0**	**3**

Course Learning Objectives:

1. To understand the basic concepts of mobile computing.
2. To learn the basics of mobile telecommunication system.
3. To be familiar with the network layer protocols and Ad-Hoc networks.
4. To know the basis of transport and application layer protocols.
5. To gain knowledge about different mobile platforms and application development.

Course Content:

UNIT I: INTRODUCTION (9 Contact hours)
Mobile Computing – Mobile Computing Vs Wireless Networking – Mobile Computing Applications – Characteristics of Mobile computing – Structure of Mobile Computing Application. MAC Protocols – Wireless MAC Issues – Fixed Assignment Schemes – Random Assignment Schemes – Reservation Based Schemes.

UNIT II: Mobile Internet Protocol And Transport Layer (9 Contact hours)
Overview of Mobile IP – Features of Mobile IP – Key Mechanism in Mobile IP – route Optimization – DHCP. Overview of TCP/IP – Architecture of TCP/IP- Adaptation of TCP Window – Improvement in TCP Performance.

UNIT III: MOBILE TELECOMMUNICATION SYSTEM (9 Contact hours)
Global System for Mobile Communication (GSM) – General Packet Radio Service (GPRS) – Universal Mobile Telecommunication System (UMTS).

UNIT IV: MOBILE AD-HOC NETWORKS (9 Contact hours)
Ad-Hoc Basic Concepts – Characteristics – Applications – Design Issues – Routing – Essential of Traditional Routing Protocols –Popular Routing Protocols – Vehicular Ad Hoc networks (VANET) – MANET Vs VANET – Security.

UNIT V MOBILE PLATFORMS AND APPLICATIONS (8 Contact hours)
Mobile Device Operating Systems – Special Constraints & Requirements – Commercial Mobile Operating Systems – Software Development Kit: iOS, Android, BlackBerry, Windows Phone.

UNIT VI: MOBILE COMMERCE (8 Contact hours)

Applications of MCommerce – Structure – Pros & Cons – Mobile Payment System – Security Issues.

Learning resources

Text book:

Prasant Kumar Pattnaik, Rajib Mall, "Fundamentals of Mobile Computing", PHI Learning Pvt. Ltd, New Delhi – 2012

Jochen Schiller, "Mobile Communications", PHI, Second Edition, 2003.

Reference Books:

Dharma Prakash Agarval, Qing and An Zeng, "Introduction to Wireless and Mobile systems", Thomson Asia Pvt Ltd, 2005.

Uwe Hansmann, Lothar Merk, Martin S. Nicklons and Thomas Stober, "Principles of Mobile Computing", Springer, 2003.

William. C. Y. Lee, "Mobile Cellular Telecommunications-Analog and Digital Systems", Second Edition, TataMcGraw Hill Edition ,2006.

C. K. Toh, "AdHoc Mobile Wireless Networks", First Edition, Pearson Education, 2002.

Course outcomes: At the end of the course, the student will be able to

CO 1	Explain the basics of mobile telecommunication systems
CO 2	Illustrate the generations of telecommunication systems in wireless networks
CO 3	Determine the functionality of MAC, network layer and Identify a routing protocol for a given Ad hoc network
CO 4	Explain the functionality of Transport and Application layers
CO 5	Develop a mobile application using android/blackberry/ios/Windows SDK
CO 6	Understand the M commerce

For Theory courses only:

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

****

Course Code	**CourseNname**	**Course category**	**L-T-P**	**Credits**
20CS3234	**Data Compression**	**PEC**	**3-0-0**	**3**

Course Objectives:

1. Define compression; understand compression as an example of representation.

2. Understand the idea of lossless and lossy compression.
3. Understand the most common file formats for image, sound and video.
4. Distinguish the basic techniques of lossless compression.

Course Content:

Unit 1: (6 Contact hours)

Introduction: Mathematical Preliminaries, Lossy and Lossless compression, Measure of performances, Application of compression, Introduction to information theory.

Unit II: (8 Contact hours)

Simple lossless Encoding, Run length encoding: Huffman coding, Applications of Huffman coding, Adaptive Huffman coding, LZW coding, LZ77 and LZ78, Run length encoding, Arithmetic coding.

Unit III: (8 Contact hours)

Fundamentals of Information Theory: Concepts of entropy, probability models, Markov models, Fundamentals of coding theory, Algorithmic information theory & Minimum description.

Unit IV: (10 Contact hours)

Lossless Compression Standards: zip, gzip, bzip, unix compression, GIF, JBIG, predictive coding

Unit V: (6 Contact hours)

Image and Video Compression: Basis functions and transforms from an intuitive point, JPEG, MPEG, Vector Quantization, Case study of WinZip, WinRar

Unit VI: (7 Contact hours)

Wavelet Based Compression, Quantization: Fundamentals of wavelets, various standard wavelet bases, Multi resolution analysis and scaling function, JPEG 2000, scaler and vector quantization.

Learning Resources

Text books:

1. Khalid Sayood, "Introduction to Data Compression", Elsevier, Fifth edition

Reference Books:

1. Mark Nelson, "The Data Compression book", BPB Publications, First edition
2. Colt McAnlis, "Understanding Compression::Data Compression for Modern Developers" PublishShroff Publishers & Distributors Pvt Ltd,First edition

Web resources:

1. https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-050j-information-and-entropy-spring-2008/videos-homework-and-readings/unit-2-lecture-1/
2. https://nptel.ac.in/courses/106102064/19
3. https://www.ics.uci.edu/~dan/pubs/DataCompression.html.

Course Outcomes: At the end of the course, the student will be able to

CO 1	Explain lossless and lossy compression and correlating them with mathematical concepts.
CO 2	Implement Huffman encoding, LZW, LZ77 and LZ78 in various projects.
CO 3	Discuss the concept of Information theory and its relation to data compression.
CO 4	Use the standards like zip, bzip etc. according to the requirements.
CO 5	Perform image and video compression according to the real world project requirements.
CO 6	Use the concepts like wavelet transform and quantization.

Assessment Method:

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

Course Code	CourseNname	Course category	L-T-P	Credits
20CS3235	**COMPUTER GRAPHICS**	**PEC**	**3-0-0**	**3**

OBJECTIVES:

1. Gain knowledge about graphics hardware devices and software used. · Understand the two dimensional graphics and their transformations.
2. Understand the three dimensional graphics and their transformations. · Appreciate illumination and color models.
3. Be familiar with understand clipping techniques.

Course Content:

UNIT I INTRODUCTION (8 Contact hours)

Survey of computer graphics, Overview of graphics systems – Video display devices, Raster scan systems, Random scan systems, Graphics monitors and Workstations, Input devices, Hard copy Devices, Graphics Software; Output primitives – points and lines, line drawing algorithms, loading the frame buffer, line function; circle and ellipse generating algorithms; Pixel addressing and object geometry, filled area primitives.

UNIT II Two Dimensional Graphics (8 Contact hours)

Two dimensional geometric transformations – Matrix representations and homogeneous coordinates, composite transformations; Two dimensional viewing – viewing pipeline, viewing coordinate reference frame; widow-to-viewport coordinate transformation, Two dimensional viewing functions; clipping operations – point, line, and polygon clipping algorithms.

UNIT III Three Dimensional Graphics (8 Contact hours)

Three dimensional concepts; Three dimensional object representations – Polygon surfaces- Polygon tables- Plane equations – Polygon meshes; Curved Lines and surfaces, Quadratic surfaces; Blobby objects; Spline representations – Bezier curves and surfaces -B-Spline curves and surfaces. TRANSFORMATION AND VIEWING: Three dimensional geometric and modeling transformations – Translation, Rotation, Scaling, composite transformations; Three dimensional viewing – viewing pipeline, viewing coordinates, Projections, Clipping; Visible surface detection methods.

UNIT IV Illumination And Colour Models (7 Contact hours)

Light sources – basic illumination models – halftone patterns and dithering techniques; Properties of light – Standard primaries and chromaticity diagram; Intuitive colour concepts – RGB colour model – YIQ colour model – CMY colour model – HSV colour model – HLS colour model; Colour selection.

UNIT V Animations & Realism 10 Animation Graphics (8 Contact hours)

Design of Animation sequences – animation function – raster animation – key frame systems – motion specification –morphing – tweening. COMPUTER GRAPHICS REALISM: Tiling the plane – Recursively defined curves – Koch curves – C curves – Dragons – space filling curves – fractals – Grammar based models – fractals – turtle graphics – ray tracing.

UNIT –VI Multimedia System: (8 Contact hours)

An Introduction, Multimedia hardware, Multimedia System Architecture. Data & File Format standards. i.e RTF, TIFF, MIDI, JPEG, DIB, MPEG,Audio: digital audio, MIDI, processing sound, sampling, compression. Video: Avi, 3GP,MOV, MPEG .

Text Books:

1. John F. Hughes, Andries Van Dam, Morgan Mc Guire ,David F. Sklar , James D. Foley, Steven K. Feiner and Kurt Akeley ,"Computer Graphics: Principles and Practice", , 3rd Edition, Addison- Wesley Professional,2013. (UNIT I, II, III, IV).
2. Donald Hearn and Pauline Baker M, "Computer Graphics", Prentice Hall, New Delhi, 2007 (UNIT V).

References:

1. Donald Hearn and M. Pauline Baker, Warren Carithers,"Computer Graphics With Open GL", 4th Edition, Pearson Education, 2010.
2. Jeffrey McConnell, "Computer Graphics: Theory into Practice", Jones and Bartlett Publishers, 2006.
3. Hill F S Jr., "Computer Graphics", Maxwell Macmillan" , 1990.
4. Peter Shirley, Michael Ashikhmin, Michael Gleicher, Stephen R Marschner, Erik Reinhard, KelvinSung, and AK Peters, Fundamental of Computer Graphics, CRC Press, 2010.
5. William M. Newman and Robert F.Sproull, "Principles of Interactive Computer Graphics", Mc GrawHill 1978. http://nptel.ac.in/

Outcomes:

CO1	Design two dimensional graphics.
CO2	Apply two dimensional transformations.
CO3	Design three dimensional graphics.
CO4	Apply three dimensional transformations.
CO5	Apply Illumination and color models.
CO6	Apply clipping techniques to graphics.
CO7	Design animation sequences.

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

LIST OF PROFESSIONAL ELECTIVES COURSE(PEC)

Elective group -4

Course Code	Course Name	Course Category	L-T-P	Credits
20CS4141	**Data Science**	**PEC**	**3-0-0**	**3**

Course Objectives:

1. To analyze, characterize empirically complex data;
2. To describe relevant statistical quantities and quantify their confidence intervals;
3. To explain sensible models and to parameterize and validate these models;
4. To quantify inter-dependency/causality structure between different variables;

Course Outcomes:

At the end of the course the students will be able to

- To develop practical data analysis skills, which can be applied to practical problems?
- To explain how math and information sciences can contribute to building better algorithms and tools.
- To develop applied experience with data science software, programming, applications and processes.

Course Content:

UNIT – I (10Contact hours)

Introduction to Data Science: Introduction to data science, exploratory data analysis, introduction to machine learning, supervised and unsupervised learning, linear regression, model selection and evolution.

UNIT-II (12 Contact hours)

Data structures, files and data plotting, Arrays & Matrices, Making Tables, Lists, Data frames, Conversion of Numeric Data frames into Matrices. Reading Excel Files, and text files, Plotting function, multiple plots, Scatter plot matrices. Basics on data acquisition, integration, cleaning, filtering, transformation.

UNIT-III (9 Contact hours)

Feature selection. Dimensionality Reduction: PCA and SVD, forward and backward feature selection, Measures of dependency. Cause and effect, spurious correlations and regularization.
Probability: Probability distributions, Marginal probability, joint probability and conditional probability. Empirical estimation of probability distributions.

UNIT–IV (9 Contact hours)

Classification Methods: Naive Bayes, Model validation: Bootstrapping & Cross Validation, Hypothesis testing, Sample t-tests, Confidence Intervals. ANOVA.

UNIT–VI (9 Contact hours)

Advanced Analytical Theory and Methods – Time Series Analysis : Overview of Time Series Analysis, ARIMA Model.

Learning resources

Text book:

1. Dunlop, Dorothy D., and Ajit C. Tamhane. '*Statistics and data analysis: from elementary to intermediate'*. Prentice Hall, 2000.

Reference Books:

1. Nina Zumel, John Mount, '*Practical Data Science with R*', Manning Publications, 2014

2. Tom Plunkett, Mark Hornick, '*Using R to Unlock the Value of Big Data: Big Data Analytics with Oracle R Enterprise and Oracle R Connector for Hadoop'*, McGraw-Hill/Osborne Media (2013), Oracle press.

Web resources:

1.Big data and analytics 17 Jul 2014 URL https://www.simplilearn.com/resources

Course outcomes: At the end of the course, the student will be able to

CO 1	Know basic notions and definitions in data analysis, machine learning.
CO 2	Know standard methods of data analysis and information retrieval
CO 3	Be able to formulate the problem of knowledge extraction as combinations of data

	filtration, analysis and exploration methods.
CO 4	Be able to translate a real-world problem into mathematical terms
CO 5	Possess main definitions of subject field
CO 6	Possess main software and development tools of data scientist, and Learn to develop complex analytical reasoning

For Theory courses only:

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

****

Course Code	**Course Name**	**Course Category**	**L-T-P**	**Credits**
20CS4142	Unixand Shell Programming	PEC	3-0-0	3

Course Learning Objectives:

1 Written technical communication and effective use of concepts and terminology.
2 Facility with UNIX command syntax and semantics.
3 Ability to read and understand specifications, scripts and programs.
4 Individual capability in problem solving using the tools presented within the class Students will demonstrate a mastery of the course materials and concepts within in class discussions.

Course Content:

UNIT I – Introduction to UNIX (8 Contact hours)

Architecture of Unix, Features of Unix , Unix Commands – PATH, man, echo, printf, script, passwd, uname, who, date, stty, pwd, cd, mkdir, rmdir, ls, cp, mv, rm, cat, more, wc, lp, od, tar, gzip,cpio,ln

UNIX Utilities: (8 Contact hours)

Process utilities, disk utilities,networking commands, unlink, du, df, mount, umount, find, unmask, ulimit, ps, w, finger, arp, ftp, telnet, rlogin.Text processing utilities and backup utilities , detailed commands to be covered are tail, head , sort, uniq, grep, egrep, fgrep, cut, paste, join, tee, pg, comm, cmp, diff, tr, awk.

UNIT II - File Management (6 Contact hours)

Introduction to unix file system, vi editor, file handling utilities, security by file permissions,File Structures, System Calls for File Management – create, open, close, read, write, lseek, link, symlink, unlink, stat, fstat, lstat, chmod, chown, Directory API – opendir, readdir, closedir, mkdir, rmdir, umask.

UNIT III – Introduction to Shells & Filters (7 Contact hours)

Introduction to Shells : Unix Session, Standard Streams, Redirection, Pipes, Tee Command, Command Execution, Command Line Editing, Quotes, Command Substitution, Job Control, Aliases, Variables, Predefined Variables, Options, Shell/Environment Customization.

Filters: Filters and Pipes, Concatenating files, Display Beginning and End of files, Cut and Paste, Sorting, Translating Characters, Files with Duplicate Lines, Count Characters, Words or Lines, Comparing Files.

UNIT IV – grep, sed, awk (7 Contact hours)

grep : Operation, grep Family, Searching for File Content.
sed : Scripts, Operation, Addresses, commands, Applications, grep and sed.
awk: Execution, Fields and Records, Scripts, Operations, Patterns, Actions, Associative Arrays, String Functions, String Functions, Mathematical Functions, User – Defined Functions, Using System commands in awk, Applications, awk and grep, sed and awk.

UNIT V – Korn shell programming (7Contact hours)

Interactive Korn Shell: Korn Shell Features, Two Special Files, Variables, Output, Input, Exit Status of a Command, eval Command, Environmental Variables, Options, Startup Scripts, Command History, Command Execution Process.

Korn Shell Programming: Basic Script concepts, Expressions, Decisions: Making Selections, Repetition, special Parameters and Variables, changing Positional Parameters, Argument Validation, Debugging Scripts, Script Examples.

UNIT VI – C shell programming (7Contact hours)

Interactive C Shell: C shell features, Two Special Files, Variables, Output, Input, Exit Status of a Command, eval Command, Environmental Variables, On-Off Variables, Startup and Shutdown Scripts, Command History, Command Execution Scripts.

C Shell Programming: Basic Script concepts, Expressions, Decisions: Making Selections, Repetition, special Parameters and Variables, changing Positional Parameters, Argument Validation, Debugging Scripts, Script Examples.

Learning resources

Text Book:

1. Behrouz A. Forouzan & Richard F. Gilberg, "Unix and Shell Programming", 1st edition, Cengage Learning, 2003

Reference Books:

1. Sumitabha Das, *"Your Unix: The Ultimate Guide"*, 1st edition, Tata Mcgraw hill, 2001
2. Graham Glass, King Ables, *"Unix for programmers and Users",* 3rd edition, Pearson Education, 2003
3. Kernighan & Pike, *"The UNIX Programming Environment",* 1st edition, Pearson Education India, 2015
4. Ken Rosen, James Farber, Rachel Klee, Douglas Host, and Dick Rosinski, *" Unix: The Complete Reference",* 2nd edition, McGraw Hill Education, 2007

Web resources:

1.Essential Linux/Unix Commands – GeeksforGeeks*https://www.geeksforgeeks.org/essential-linuxunix-commands/*
2.Introduction to Linux Shell and Shell Scripting - GeeksforGeeks
https://www.geeksforgeeks.org/introduction-linux-shell-shell-scripting/
3. UNIX /LINUX TUTORIAL
https://www.tutorialspoint.com/unix/

Course outcomes: At the end of the course, the student will be able to

CO 1	Documentation will demonstrate good organization and readability.
CO 2	Able to do Data organization, problem solving and research.
CO 3	Demonstrate simple effective user interfaces.
CO 4	Demonstrate effective use of structured programming.
CO 5	Will be accompanied by printed output demonstrating completion of a test plan

Assessment Method:

Course Nature	**Theory**

Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

****

Course Code	Course Name	Course Category	L-T-P	Credits
20CS4143	VLSI	PEC	3-0-0	3

Course Objectives

1. To make understand the student the introductory concepts of Very Large Scale Integrated Circuits design such as HDLs, CMOS circuit design, Layout techniques and fabrication

Course Content:

Unit-I **(13.5 Contact hours)**

Finite State Machines and Verilog HDL

Mealy state machines, Moore state machines, Conversion of Mealy machines to Moore machines and vice-versa, Redundancy elimination, Sequence detectors with and without overlapping, Parity detectors and generators.

Verilog HDL: Gatelevel modeling, Dataflow modeling, Behavioral modeling, Dealy modeling, Writing test bench, FSM modeling.

Unit-II **(7.5 hours)**

Digital sytem Design and Verilog HDL

Specifications, Moore's Law, Data path design, Control path design, GCD Datapath design, GCD Control path design, State machine for traffic light controller . FPGA Design flow, ASIC Design flow.

Verilog HDL: Datapath, Control path, System Design

Unit –III **(3 hours)**

Fabrication

Fabrication process flow: cleaning, oxidation, patterning, Mask Alignment, Lithography- types, etching-types, annealing, different photo-resists, Electrical and Mechanical Testing, Cleaning

Unit –IV (3 hours)

CMOS Circuit Design

Design of inverter, NAND, NOR, EXOR, Half-Adder, Full adder, Multiplexer and other circuits using CMOS logic

Unit –V (3 hours)

Circuit Layout Design

MOS Layers, Stick Diagrams, Design Rules and Layout, General Observations on the Design rules, Layout diagrams and optimization techniques

LearningResourceTex

t Books

1. Stephen Brown, ZvokoVranesic, *'Fundamentals of Digital Design using Verilog'*, Mc Graw Hill publications
2. Douglas A.Pucknell and Kamran Eshraghian, *'Basic VLSI Design'*, PHI publications

Reference Books

1. Samir Palnitkar, *'Verilog HDL - A Guide to Digital Design and Synthesis'*, Pearson publications
2. Stephen D. Senturia, *'Microsystem Design'*, Kluwer Academic Publishers
3. Marc Madou, *'Fundamentals of Microfabrication*', CRC Press
4. Ian Grout, *'Digital Systems Design with FPGAs and CPLDs'*,Elsevier-2008

Web Resources

1. Prof S Shankar Balachandran, NPTEL-IIT Madras, *'Digital circuits & Systems'*. URL: http://nptel.ac.in/courses/117106114/
2. Prof S Srinivasan, NPTEL - IIT Madras, *'Digital circuits and systems*' URL: https://nptel.ac.in/courses/117106086/
3. Prof S Parasuraman, NPTEL-IIT Madras, *'Electronic materials, devices and fabrication'*. URL: https://nptel.ac.in/courses/113106062/
4. Deepak Kumar Tala, URL: http://www.asic-world.com

Course Outcomes:

At the end of the course, the student will be able to

CO 1	Understand specifications of VLSI designs, Moore's Law
CO 2	Different VLSI Design flows - FPGA, ASIC
CO 3	Understand the concepts of Finite State Machines and its relevance in IC Design
CO 4	Modeling of digital designs using hardware description language
CO 5	Understand the different steps involved in IC fabrication process
CO 6	Understand the concept of IC layout

Assessment Method

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

Course Coded	**Course Name**	**Course Category**	**L-T-P**	**Credits**

20CS4144	**Soft Computing**	**PEC**	**3-0-0**	**3**

Course Learning Objectives:

Students should be able to

1. To learn about soft computing techniques and their applications.
2. To analyse various neural network architectures.\
3. To define fuzzy system
4. To understand the genetic algorithm concepts and their applications.
5. To identify and select a suitable soft computing technology to solve the problem, construct a solution and implement a soft computing solution
6. To learn about hybrid systems.

Course Content:

UNIT I (Introduction to Soft Computing) (7 Contact hours)

soft computing,Hard computin**g,** Artificial neural networks-biological neurons, Basic models of artificial neural networks- Connections, Learning, Activation Functions, McCulloch and Pitts Neuron.

UNIT II (Perceptron Networks) (7 Contact hours)

Learning rule-training algorithm, multiple adaptive linear neuron, Adaptive Linear Neuron, Back propagation Network-Architecture, Applications of ANN,single layer perceptron,multilayer perceptron.

UNIT III (Fuzzy Logic) (7 Contact hours)

Fuzzy sets-properties-operations on fuzzy sets, fuzzy relations- operations on fuzzy relations,crisp logic,fuzzy logic,,Differences of fuzzy and crisp logic.

UNIT IV (Fuzzy membership functions) (7Contact hours)

Fuzzification, Methods of membership value assignments- intuition- inference-rank ordering, Lambda-cuts for fuzzy sets, Defuzzification methods,

UNIT V (Truth values and fuzzy logic) (8 Contact hours)

Fuzzy propositions, Formation of fuzzy rules- Decomposition of rules –Aggregation of r ules, Fuzzy Interface Systems – Mamdani and Surgeon types, Neuro-fuzzy hybrid systems – characteristics- Classification.

UNIT VI (Genetic Algorithm) (8 Contact hours)

Introduction to genetic algorithm, operators in genetic algorithm – coding – selection – cross over- Find Mutation, stopping condition for genetic algorithm flow, Genetic- neurohybrid systems, Genetic-Fuzzy rule-based system.

Learning resources:

Text books:

1. S.N.Sivanandam and S.N.Deepa, "*Principles of soft computing*", Wiley India.
2. Timothy J.Ross, "*Fuzzy Logic with engineering applications",* Wiley India.

Reference Books:

1. N.K. Sinha and M.M Gupt, *Soft Computing & Intelligent Systems: Theory & Applications*- Academic Press /Elsevier.2009.
2. Simon Haykin, *Netural Network- A Comprehensive Foundation*- Prentice Hall International,Inc.
3. R.Eberhart and Y. Shi,*Computational Intelligence: Concepts to Implementation,* Morgon Kaufman/Elsevier,2007.
4. Drainkov T.j. "*Fuzzy Logic With Engineering Applications"* McGraw Hill.
5. Bart KOsko, "*Neural Network and Fuzzy Systems*-Prentice"Hall,Englewood Cliffs.
6. Goldberg D.E, "*Genetic Algorithms in Search,Optimization,and Machine Learing*

Web resources:

1. https://www.scribd.com/document/250449108/Definition-of-Soft-Computing
2. https://bookboon.com/en/introduction-to-soft-computing-ebook

Course outcomes: At the end of the course, the student will be able to

CO 1	Knowledge about fuzzy logic and concept of fuzziness involved in various systems and fuzzy set theory.
CO 2	Knowledge about fuzzy sets, representing fuzzy rules, approximate reasoning, fuzzy inference systems and fuzzy logic,
CO 3	Knowledge about fundamental theory of neural networks, neural network architectures, algorithms, applications and their limitations.
CO 4	Knowledge for appropriate learning rules for each of the architectures and learn several neural network paradigms and its applications.
CO 5	Knowledge of revealing different applications of these models to solve engineering and other problems.

For Theory courses only:

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

Course Code	Course Name	Course Category	L-T-P	Credits
20CS4145	**File Structures**	PEC	3-0-0	3

Course Learning Objectives:

1. Understand file structures including sequential, indexed, indexed sequential, hashed file structures
2. Apply object-oriented concepts to design file systems
3. Understand B^+-trees to implement file systems
4. Implement file operations including read, write, update and search
5. Develop and analyse external sorting methods

UNIT-I **(8 Contact hours)**

Fundamental File Structure Concepts: Field and Record Organization, Using Classes to Manipulate Buffers, Using Inheritance for Record Buffer Classes, Managing Fixed Length, Fixed Field Buffers, An Object-Oriented Class for Record Files.

Managing Files and Records: Record Access, More about Record Structures, Encapsulating Record I/O Operations in a Single Class, File Access and File Organization.

UNIT-II **(8 Contact hours)**

Fundamental File Processing Operations: Physical Files and Logical Files, Opening Files, Closing Files, Reading and Writing, Seeking, Special Characters in Files, The UNIX Directory

Structure, Physical Devices and Logical Files, Physical Devices as Files ,File-related Header Files, UNIX File System Commands.

Indexed Files of Data Objects- Indexing: A Simple Index for Entry-Sequenced File, Template Classes in C++, Object-Oriented support for Indexed, Entry-Sequenced Files of Data Objects, Indexes That Are Too Large to Hold in Memory, Indexing to Provide Access by Multiple Keys, Retrieval Using Combinations of Secondary Keys, Improving the Secondary Index Structure: Inverted Lists, Selective Indexes, Binding.

UNIT-III (8 Contact hours)

Multilevel Indexing and B-Trees: Introduction: The Invention of the B-Tree, Statement of the Problem, Indexing with Binary Search Trees, Multi-level Indexing, A Better Approach to Tree Indexes, B-Trees: Working up from the Bottom, Example of Creating a B-Tree, An Object-Oriented Representation of B-Trees, B-Tree Methods Search, Insert, and Others, B-Tree Nomenclature, Formal Definition of B-Tree Properties, Worst-case Search Depth, Deletion, Merging, and Redistribution, Redistribution during Insertion: A Way to Improve Storage Utilization, B* Trees, Buffering of Pages: Virtual B-Trees, Variable-length Records and Keys.

UNIT-IV (8 Contact hours)

Indexed Sequential File Access and B+ Trees : Indexed Sequential Access, Maintaining a Sequence Set, Adding a Simple Index to the Sequence Set, The Content of the Index: Separators Instead of Keys, The Simple Prefix B+ Tree, Simple Prefix B+ Tree Maintenance, Index Set Block Size, Internal Structure of Index Set Blocks: A Variable-order B-Tree, Loading a Simple Prefix B+ Tree, B+ Trees, B-Trees, B+ Trees, and Simple Prefix B+ Trees in Perspective.

UNIT-V (8 Contact hours)

Hashing: Introduction, A Simple Hashing Algorithm, Hashing Functions and Record Distributions, How Much Extra Memory Should Be Used, Collision Resolution by Progressive Overflow, Storing More Than One Record per Address: Buckets, Making Deletions, Other Collision Resolution Techniques, Patterns of Record Access.

UNIT-VI (7 Contact hours)

Extendible Hashing: Introduction, How Extendible Hashing Works, Implementation, Deletion, Extendible Hashing Performance, Alternative Approaches, Multi list and Inverted Files, Sorting of Large Files.

External sorting: Secondary storage algorithms.

Text Book

1. Michael j. Folk, Greg Riccardi, Bill Zoellick; *File Structures: An Object Oriented Approach with C++*, 3/e Pearson Publishers.
2. Suggested Reading

Course outcomes: At the end of the course, the student will be able to

CO 1	Understand file structures including sequential, indexed, indexed sequential, hashed file structures
CO 2	Apply object-oriented concepts to design file systems
CO 3	Implement file operations including read, write, update and search
CO 4	Develop and analyze external sorting methods

For Theory courses only:

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

****

LIST OF PROFESSIONAL ELECTIVES COURSE(PEC)
ELECTIVE GROUP-5

Course Code	Course Name	Course Category	L-T-P	Credits
20CS4251	Optimization Techniques	PEC	3-0-0	3

Course Learning Objectives:

1. To define an objective function and constraint functions in terms of design variables, And then state the optimization problem.
2. To state single variable and multi variable optimization problems, without and with Constraints.
3. To explain linear programming technique to an optimization problem, define slack And surplus variables, by using Simplex method.
4. To state transportation and assignment problem as a linear programming problem to Determine Simplex method.
5. To study and explain nonlinear programming techniques, unconstrained or
6. Constrained, and define exterior and interior penalty functions for optimization problems.
7. To explain Dynamic programming technique as a powerful tool for making a Sequence of interrelated decisions.

Course Content:

UNIT – I **(8 Contact hours)**

Introduction and Classical Optimization Techniques: Statement of an Optimization problem – design vector – design constraints – constraint surface – objective function – objective function surfaces – classification of Optimization problems.

UNIT – II **(8 Contact hours)**

Classical Optimization Techniques : Single variable Optimization – multi variable Optimization without constraints – necessary and sufficient conditions for minimum/maximum – multivariable Optimization with equality constraints. Solution by method of Lagrange multipliers – multivariable Optimization with inequality constraints – Kuhn – Tucker conditions.

UNIT – III **(8 Contact hours)**

Linear Programming : Standard form of a linear programming problem – geometry of linear programming problems – definitions and theorems – solution of a system of linear simultaneous equations – pivotal reduction of a general system of equations – motivation to the simplex method – simplex algorithm – Duality in Linear Programming – Dual Simplex method.

UNIT – IV **(7 Contact hours)**

Transportation Problem : Finding initial basic feasible solution by north – west corner rule, least cost method and Vogel's approximation method – testing for optimality of balanced transportation problems – Special cases in transportation problem.

UNIT – V **(7 Contact hours)**

Nonlinear Programming: Un constrained cases – One – dimensional minimization methods: Classification, Fibonacci method and Quadratic interpolation method – Univariate method, Powell's method and steepest descent method.

Constrained cases – Characteristics of a constrained problem, Classification, Basic approach of Penalty Function method; Basic approaches of Interior and Exterior penalty function methods. Introduction to convex Programming Problem.

UNIT – VI **(7 Contact hours)**

Dynamic Programming: Dynamic programming multistage decision processes – types – concept of sub optimization and the principle of optimality – computational procedure in dynamic programming – examples illustrating the calculus method of solution – examples illustrating the tabular method of solution.

Learning resources

Text books:

1. S.S.Rao - "Engineering optimization : Theory and practice", New Age International (P) Limited, 3rd edition, 1998.
2. H.S. Kasene & K.D. Kumar "Introductory Operations Research" , Springer (India), Pvt. LTd.

Reference Books

1. K.V. Mital and C. Mohan -"Optimization Methods in Operations Research and systems Analysis" , New Age International (P) Limited, Publishers, 3rd edition, 1996.
2. Dr. S.D.Sharma, Kedarnath, Ramnath & Co -Operations Research
3. G. Hadley "Operations Research : An Introduction" – by H.A.Taha,PHI pvt. Ltd., 6th edition Linear Programming.

Course outcomes: At the end of the course, the student will be able to

CO 1	State and formulate the optimization problem, without and with constraints, by using design variables from an engineering design problem
CO 2	Apply classical optimization techniques to minimize or maximize a multi-variable objective function, without or with constraints, and arrive at an optimal solution.
CO 3	Formulate a mathematical model and apply linear programming technique by using Simplex method. Also extend the concept of dual Simplex method for optimal solutions
CO 4	Solve transportation and assignment problem by using Linear programming Simplex method.
CO 5	Apply gradient and non-gradient methods to nonlinear optimization problems and use interior or exterior penalty functions for the constraints to derive the optimal solutions.
CO 6	Formulate and apply Dynamic programming technique to inventory control, production planning, engineering design problems etc. to reach a final optimal solution from the current optimal solution.

For Theory courses only:

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

****

Course Code	**Course Name**	**Course Category**	**L-T-P**	**Credits**

20CS4252	DESIGN PATTERNS	PEC	3-0-0	3

Course Objectives

1. The focus of this course is on design rather than implementation.
2. Introducing the Unified Process and showing how UML can be used within process.
3. Presenting a comparison of the major UML tools for industrial-strength development.
4. Introduction to design patterns, practical experience with a selection of central patterns.

Course Content: **(8 Contact hours)**

Unit I Introduction : Introduction to OOAD; typical activities / workflows / disciplines in OOAD, Introduction to iterative development and the Unified Process, Introduction to UML; mapping disciplines to UML artifacts, Introduction to Design Patterns – goals of a good design, Introducing a case study & MVC architecture.

Unit II Inception: **(8 Contact hours)**

Artifacts in inception, Understanding requirements – the FURPS model, Understanding Use case model – introduction, use case types and formats, Writing use cases – goals and scope of a use case, elements / sections of a use case, Use case diagrams, Use cases in the UP context and UP artifacts, Identifying additional requirements, Writing requirements for the case study in the use case model.

Unit III Elaboration: **(8 Contact hours)**

System sequence diagrams for use case model, Domain model : identifying concepts, adding associations, adding attributes, Interaction Diagrams, Introduction to GRASP design Patterns ,Design Model: Use case realizations with GRASP patterns, Design Class diagrams in each MVC layer Mapping Design to Code, Design class diagrams for case study and skeleton code

Unit IV More Design Patterns: **(7 Contact hours)**

Fabrication, Indirection, Singleton, Factory, Facade, Publish-Subscribe

Unit V More UML diagrams : **(7 Contact hours)**

State-Chart diagrams, Activity diagrams, Component Diagrams, Deployment diagrams, Object diagrams.

Unit VI Advanced concepts in OOAD : **(7 Contact hours)**

Use case relationships, Generalizations Domain Model refinements, Architecture, Packaging model elements.

Course Outcomes

1. Students successfully completing this course will be able to:
2. identify the purpose and methods of use of common object-oriented design patterns
3. select and apply these patterns in their own designs for simple programs
4. represent the data dependencies of a simple program using UML
5. represent user and programmatic interactions using UML
6. create design documentation outlining the testable and complete design of a simple program
7. produce and present documents for the purpose of capturing software requirements and specification
8. produce plans to limit risks specific to software designed for use in a particular social context

Text books

1. Applying UML and patterns' by Craig Larman, Pearson

2. Object-Oriented Analysis & Design with the Unified Process by Satzinger, Jackson & Burd Cengage Learning
3. ‘UML distilled’ by Martin Fowler , Addison Wesley, 2003
4. Reference
5. O’reilly ‘ s ‘Head-First Design Patterns’ by Eric Freeman et al, Oreillly
6. UML 2 Toolkit, by Hans-Erik Eriksson, Magnus Penker, Brian Lyons, David Fado:WILE\’- Dreamtech lndia Pvt. Lid.

For Theory courses only:

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

****

Course Code	**Course Name**	**Course Category**	**L-T-P**	**Credits**
20CS4253	**CLOUD COMPUTING**	**PEC**	**3-0-0**	**3**

Course Learning Objectives:

1. Understand various basic concepts related to cloud computing technologies. 2.Understand the architecture and concept of different cloud models: IaaS, PaaS, SaaS.
2. Understand big data analysis tools and techniques.
3. Understand of cloud virtualization, cloud storage, data management and data visualization. Understand different cloud programming platforms and tools.

Course Content:

UNIT-I **(9 Contact hours)**

Introduction to Cloud Computing: Cloud Computing in a Nutshell, System Models for Distributed and Cloud Computing, Roots of Cloud Computing, Grid and Cloud, Layers and Types of Clouds, Desired Features of a Cloud, Basic Principles of Cloud Computing, Challenges and Risks, Service Models .

UNIT-II **(9 Contact hours)**

Virtual Machines and Virtualization of Clusters and Data Centers, Levels of Virtualization, Virtualization Structures / tools and Mechanisms, Virtualization of CPU, Memory and I/O Devices, Virtual Clusters and Resource Management, Virtualization Data-Centre Automation.

UNIT-III **(7 Contact hours)**

Cloud computing architectures: over Virtualized Data Centers: Data–Center design and Interconnection networks, Architectural Design of Compute and Storage Clouds, Public Cloud Platforms, GAE, AWS, Azure, Inter-cloud Resource Managemen.

UNIT-IV **(8 Contact hours)**

Cloud Programming and Software Environments: Features of Cloud and Grid Platforms, parallel and distributed Programming Paradigms, Programming Support of Google App Engine, Programming on Amazon AWs and Microsoft Azure, Emerging Cloud Software Environments.

UNIT-V **(7 Contact hours)**

Disaster Recovery,Disaster Recovery Planning, comparing approaches in Xen,Eucalyptus,OpenNebula,Amazon,Nimbus.

UNIT-VI **(7 Contact hours)**

Various trends in computing,web services,service oriented architecture(SOA)

Learning resources

Text book:

1. John W. Rittenhouse, James F. Ransome, "Cloud Computing: Implementation, Management, and Security ", CRC Press, 2009.
2. RajkumarBuyya, James Broberg, Andrzej M. Goscinski, "Cloud Computing:
3. Principles and Paradigms", WileyPublishing, 2011.
4. Sandeep Bhowmik "Cloud Computing" Cambridge Publishing, 2017

Reference Books:

1. Morgan Kaufmann, '*Distributed and Cloud Computing*', 1st edition, 2011.
2. Gautam Shroff,' *Enterprise Cloud Computing*', Cambridge, 2010.
3. Ronald Krutz and Russell Dean Vines, '*Cloud Security'*, 1st Edition, Wiley, 2010.

Course outcomes: At the end of the course, the student will be able to

CO 1	Develop and deploy cloud application using popular cloud platforms,
CO 2	Design and develop highly scalable cloud-based applications by creating and configuring virtual machines on the cloud and building private cloud.
CO 3	Explain and identify the techniques of big data analysis in cloud.
CO 4	Identify the technological drivers of cloud computing paradigm.
CO 5	Identify the security issues in cloud computing
CO 6	Develop cloud services using popular cloud platforms

For Theory courses only:

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

****

Course code	Course name	Course Category	L-T-P	Credits
20CS4254	BLOCK CHAIN TECHNOLOGY	PEC	3-0-0	3

Course Outcomes:

1. Understand the architectural components of a block chain system
2. Understand the inner workings of smart contracts as means for developing decentralized applications
3. understand the details of interactions between the enclosed smart contract network and the external world, be aware of further implications these interactions pose to the aspect of decentralization

UNIT-1

Introduction to Block chain: The story of a transaction, From Transactions to Blocks, Blocks and Distributed Consensus, Design Primitives Protocols, Security, Consensus, Permissions, Privacy Block chain Architecture and Design Basic crypto primitives of Hash, Signature, Hash chain to Block chain Basic mechanisms, Introduction to major block chain platforms.

UNIT-2

Development environments in block chain: Requirements for the consensus protocols, Proof of Work, Scalability aspects of Block chain consensus protocols, Permission Block chains Design goals; Block chain deployment, Mining and forking, Segregated Witness Block chain architectures-Abstract Architecture, Introduction to major block chain platforms.

UNIT-3

Block chain in Multitude of clients in Ethereum, Production and test networks in Ethereum, Public, private and development deployments; Solidity in depth, Building blocks popular contracts already in deployment; Consensus protocols for Permissioned Block chains; Hyperledger Fabric I: Decomposing the consensus process, Hyperledger fabric components, Chain code Design and Implementation; Hyperledger Fabric II: Beyond Chain code: fabric SDK and Front End Hyperledger composer tool.

UNIT- 4

Block chain in Financial Software and Systems Settlements KYC, Capital markets, Insurance Block chain in trade/supply chain, and Provenance o Block chain for Government: Digital identity, land records and other kinds of public distribution system / social welfare systems; Conceptual distinction between a payment system and a decentralized applications platform. Differences in their architectures from security-first aspect to a rich feature set, Future roadmap for them, following their own paths with probable interconnections.

UNIT-5

Block chain Cryptography: Privacy and Security on Block chain, Research aspects: Secure cryptographic protocols on Block chain; Secured Multi-party Computation;

UNIT-6

Block chain for science making better use of the data-mining network; Considerations for production deployment a) Quality of decentralized applications in Code patterns, Security Other smart contract platforms, Discussion of future prospects.

Text Books

Robert C. Hackney, "Lawyer's Guide to Block chain Technology: What it is and how it will disrupt the practice of law", Kindle Edition, O'Reilly Publishing, 2017.

Reference Books

- Andreas Antonopoulos, "Mastering Bit Coin", O'Reilly Publishing, 2014.
- Stephan Fleming, "Block chain Technology & Micro services Architecture: A Non-Programmer's Handbook "Kindle Edition, Neha Dubey, 2018.

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

****

Course Code	Course Name	Course Category	L-T-P	Credits
20CS4255	**INTERNET OF THINGS**	**PEC**	**3-0-0**	**3**

Course Outcomes:

- Familiarity with the essential protocols of IoT and their operations
- Design and implementation of IoT networks
- Identifying various design parameters for developing IoT applications
-

UNIT-1

Introduction: Definition and functional requirements, History of IoT, Architecture, Major components of IOT devices, Enabling technologies of IoT, Four pillars of IoT, DNA of IoT, Overview of wireless communication technologies, Internet and Web 3.0 View of IoT, Ubiquitous IoT applications, Toolkit approach for End-user participation in IoT.

UNIT-2

Programming the Microcontroller for IoT: Basics of Sensors and actuators, Examples and working principles of sensors and actuators, Arduino/Equivalent microcontroller platform, Setting up the board, Programming for IoT; Reading from Sensors Communication: Connecting microcontroller with mobile devices, Communication through bluetooth and USB, Connection with the internet using wi-fi/ethernet.

UNIT-3

IoT-Middleware and Protocols**:** Overview of middleware, Communication middleware for IoT, LBS and surveillance middleware, IoT information security, Protocol standardization for IoT, Efforts, M2M and WSN protocols, SCADA and RFID protocols, Issues with IoT standardization, Unified data standards, Protocols, IEEE 802.15.4, BACNet protocol, Modbus, KNX, Zigbee,Zwave,6LOWPAN, Architecture, Network layer, APS layer, Security.

UNIT-4

Web of Things:WoTvs. IoT, Two pillars of the Web, Architecture Standardization for WoT, Platform middleware for WoT, Unified Multitier WoT architecture, WoT portals and business intelligence.

UNIT - 5:

Cloud of Things: Grid/SOA and cloud computing, Cloud middleware, Cloud standards, Cloud providers and systems, IoT and cloud computing, Set up cloud environment - Send data from microcontroller to cloud, Mobile cloud computing, Cloud of things architecture.

UNIT-6

IoT Physical Servers and Cloud Offerings: Introduction to Cloud Storage models and communication APIs, Webserver – Web server for IoT, Cloud for IoT, Python web application framework, Designing a RESTful web API.

Text Book:

Honbo Zhou, "The Internet of Things in the Cloud: A Middleware Perspective", CRC Press, 2012.

Reference Books:

4. Hakima Chaouchi, "The Internet of Things, Connecting objects to the web", ISTE & Wiley Publications, 2010.
5. Dieter Uckelmann, Mark Harrison, Florian Michahelles, "Architecting the Internet of Things", Springer Science & Business Media, 2011.
6. Charalampos Doukas, "Building Internet of Things with the Arduino", Create space, 2002.
7. **Jan Holler, Vlasios Tsiatsis, Catherine Mulligan, Stefan Aves and, Stamatis Karnouskos, David Boyle, "From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence", 1st Edition, Academic Press, 2014.**

****

Course Code	Course Name	Course category	L-T-P	Credits
20CS4257	**Computer Vision**	**PEC**	**3-0-0**	**3**

Course Learning Objectives:

Students undergoing this course are expected to:

1. To review image processing techniques for computer vision
2. To understand Image Homographies
3. To understand Scale Invariant Feature Transform and its applications
4. To understand Camera Model (Intrinsic and Extrinsic Parameters) and Epipolar geometry
5. To understand motion analysis
6. To study some applications of computer vision algorithms

Course Content:

Unit- I: (5 Contact Hours)

Introduction to Computer vision, Review of Linear Algebra, Review of Probability, Introduction to OPENCV/Matlab.

Unit- II: (6 Contact Hours)

Basic relations b/w pixels, Linear filters, Convolutions and correlation, Finite difference filters, Smoothing filters, box filter, Guassian image noise model, Gaussian smoothing filter,Gradients and Edge detection,canny edge detector, Smoothing and differentiation, Prewitt filter, sobel filter, Derivative of Gaussian filter, Types of edges: step, ramp, ridge. , canny edge detector.

Unit - III: (8 Contact Hours)

Harris corner detection, feature matching, Image mappings, Planar Hymnographies, Similarity transformations, Affine transformations, warping images, SIFT, RANSAC, creating panoramas.

Unit - IV: (10 Contact Hours)

Correspondence problem, correspondence matching, Camera model, Pinhole camera model, camera calibration, Augmented reality, Epipolar geometry, Epipoles, Epipolar lines, Conjugate Epipolar lines, computing with cameras and 3D structures, Multiple view reconstruction, Stereo images , recover depth from two images, geometry of simple stereo system, stereo disparity, Image pyramids, Camera projection: (Extrinsics and Intrinsics parameters),

Unit - V: (**8 Contact Hours)**

Optical flow, Flow estimation, Brightness constancy equation, computing optic flow, Horn Shunck and Lucas Kanade Algorithms, Pyramids, Video change detection, Background subtraction, frame differencing, Adaptive Background subtraction, Presistant frame differencing, Appearance based tracking, Video tracking : Mean-shift and Lucas-kanade tracking .

Unit -VI: (8 Contact Hours)

Image clustering and classification, k-means clustering, Hierarchical clustering, spectral clustering, Image segmentation, Graph cuts, Segmentation using clusters, variational methods, Visual tracking, Intro to Deep learning.

Learning resources

Text Books:

1. D. Forsyth and J. Ponce, "Computer Vision - A modern approach", McGraw-Hill.

Reference Books:

1. Richard Szeliski ,"Computer Vision: Algorithms and Applications"
2. Fisher et al , "Dictionary of Computer Vision and Image Processing"
3. Simon J.D. Prince, "Computer Vision: Models, Learning, and Inference"

Web References:

1. University of Central Florida, Sept 2012, "Computer Vision" https://www.youtube.com/watch?v=715uLCHt4jE&list=PLd3hlSJsX_Imk_BPmB_H3AQjFKZS9XgZm
2. University of Pennsylvania, FALL 2007, " Introduction to Computer Vision",
   a. http://www.cse.psu.edu/~rtc12/
3. Stanford University, Fall 2016-17, " Computer Vision : Foundations and Applications"
4. http://vision.stanford.edu/teaching/cs131_fall1617/schedule.html

Course outcomes: At the end of the course, the student will be able to

CO 1	identify basic concepts, terminology, theories, models and methods in the field of computer vision
CO 2	describe basic methods of computer vision related to multi-scale representation, edge detection
CO 3	detection of other primitives, stereo, motion and object recognition,
CO 4	suggest a design of a computer vision system for a specific problem

Assessment Method:

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

LIST OF OPEN ELECTIVES COURSE (OEC) FOR ALL BRANCHES

Course Code	Course Name	Course Category	L-T-P	Credits
20CS4261	Big Data Analytics	PEC	3-0-0	3

Course Learning Objectives:

1. Applying and understanding the big data flow for the actual projects.
2. Understands the lifecycle of the data analytics & big data ecosystem and able to apply for real world problems.
3. Acquires knowledge on the tools and techniques for solving big data analytics.
4. Learns how to apply the mining techniques on big data

Course Content:

Unit - I (8 Contact hours)

Introduction to Big Data Analytics:Big Data Overview, State of the Practice in Analytics, Key Roles for the New Big Data Ecosystem, Examples of Big Data Analytics.

Data Analytics Lifecycle:Data Analytics Lifecycle Overview, Discovery, Data Preparation, Model Planning, Model Building, Communicate Results, Operationalize.

Unit - II (8 Contact hours)

Advanced Analytical Theory and Methods- Clustering: Overview of Clustering, K-means, Additional Algorithms Advanced Analytical Theory and Methods-Association Rules: Overview, Apriori Algorithm, Evaluation of Candidate Rules, Applications of Association Rules, An Example: Transactions in a Grocery Store, Validation and Testing, Diagnostics.

Unit-III (8 Contact hours)

Advanced Analytical Theory and Methods- Regression: Linear Regression, Logistic Regression, Reasons to Choose and Cautions, Additional Regression Models Advanced Analytical Theory and Methods-Classification: Decision Trees, Naïve Bayes, Diagnostics of Classifiers, Additional Classification Methods

Unit – IV (8 Contact hours)

Advanced Analytical Theory and Methods-Time Series Analysis: Overview of Time Series Analysis, ARIMA Model, Additional Methods.

Advanced Analytical Theory and Methods-Text Analysis: Text Analysis Steps, A Text Analysis Example, Collecting Raw Text, Representing Text, Term Frequency--Inverse Document Frequency (TFIDF), Categorizing Documents by Topics, Determining Sentiments, Gaining Insights.

UNIT V: (7 Contact hours)

Advanced Analytics--Technology and Tools-MapReduce and Hadoop:Analytics for Unstructured Data, The Hadoop Ecosystem, NoSQL.Advanced Analytics--Technology and Tools-In-Database Analytics: SQL Essentials, In-Database Text Analysis, Advanced SQL.The Endgame or Putting It All Together:Communicating and Operationalizing an Analytics Project, Creating the Final Deliverables, Data Visualization Basics.

UNIT – VI: (7 Contact hours)

Social Media Analytics and Text Mining: Introducing Social Media; Key elements of Social Media; Text mining; Understanding Text Mining Process; Sentiment Analysis, Performing Social Media Analytics and Opinion Mining on Tweets;

Learning resources

Text book:

1. Dunlop, Dorothy D., and Ajit C. Tamhane. '*Statistics and data analysis: from elementary to intermediate'*. Prentice Hall, 2000.

2. EMC Education Services "Data Science and Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data" Wiley Publishers.
3. Tom White " Hadoop: The Definitive Guide" Third Edition, O"reilly Media, 2011.
4. Prajapati, "V. Big data analytics with R and Hadoop",Packt Publishing Ltd, 2013.

Reference Books:

1. Ohlhorst, Frank J. *'Big data analytics: turning big data into big money'*. John Wiley & Sons, 2012.
2. W.N. Venables, D.M Smith, *'An introduction to R'*, Network Theory Ltd.
3. Nina Zumel, John Mount, '*Practical Data Science with R*', Manning Publications, 2014
4. Tom Plunkett, Mark Hornick, '*Using R to Unlock the Value of Big Data: Big Data Analytics with Oracle R Enterprise and Oracle R Connector for Hadoop'*, McGraw-Hill/Osborne Media (2013), Oracle press.

Web resources:

1. Implementing big data analysis, 19 Dec 2017 URL https://aiodex.com/?ref=5b45a599c7165
2. Big data and Hadoop introduction 17 Jul 2014 URL http://www.simplilearn.com/big-data-a
3. Big data and analytics 17 Jul 2014 URL https://www.simplilearn.com/resources

Course outcomes: At the end of the course, the student will be able to

CO 1	Have a clear idea about the big data flow and its ecosystem**.**
CO 2	Apply the tools and techniques on big data while applying data mining techniques**.**
CO 3	Use statistical tool and statistical methods that can be applied on big data.
CO 4	Have a clear idea about how to represent the unstructured data in the data bases
CO 5	Understand the common Hadoop ecosystem components, Hadoop Architecture, HDFS, Anatomy of File Write and Read, Rack Awareness.
CO 6	Have a clear idea about social media data,text mining,mobile analytics.

For Theory courses only:

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

****

****

Course Code	**Course Name**	**Course Category**	**LTP**	**Credits**
20CS4262	**BIO-METRIC SECURITY**	**PEC**	**3-0-0**	**3**

Course Educational Objective:

1. To provide students with understanding of biometrics, biometric equipment and standards applied to security.

Course Content:

UNIT-I **(8 Contact hours)**

Biometrics- Introduction- benefits of biometrics over traditional authentication systems -benefitsof biometrics in identification systems-selecting a biometric for a system –Applications - Key.biometric terms and processes - biometric matching methods -Accuracy in biometric systems.

UNIT-II **(8 Contact hours)**

Physiological Biometric Technologies: Fingerprints - Technical description –characteristics - Competing technologies - strengths – weaknesses – deployment - Facial scan - Technical description - characteristics - weaknesses-deployment - Iris scan - Technical description – characteristics - strengths – weaknesses – deployment - Retina vascular pattern.

UNIT-III **(8 Contact hours)**

Technical description – characteristics - strengths – weaknesses – deployment - Hand scan - Technical description-characteristics - strengths – weaknesses deployment – DNA biometrics.

UNIT-IV **(7 Contact hours)**

Behavioral Biometric Technologies: Handprint Biometrics - DNA Biometrics.

UNIT-V **(7 Contact hours)**

signature and handwriting technology - Technical description – classification – keyboard / keystroke dynamics- Voice – data acquisition - feature extraction - characteristics - strengths – weaknesses-deployment.

UNIT-VI **(7 Contact hours)**

Multi biometrics and multi factor biometrics - two-factor authentication with passwords - tickets and tokens – executive decision - implementation plan.

Course Outcomes:

1. 1.Demonstrate knowledge of the basic physical and biological science and engineering principles underlying biometric systems.
2. 2.Understand and analyze biometric systems at the component level and be able to analyze and design basic biometric system applications.
3. 3.Be able to work effectively in teams and express their work and ideas orally and in writing.
4. 4.Identify the sociological and acceptance issues associated with the design and implementation of biometric systems**.**
5. 5.Understand various Biometric security issues.

TEXT BOOKS:

1. 1.Samir Nanavathi, Michel Thieme, and Raj Nanavathi : “Biometrics-Identity verification in a network”, 1st Edition, Wiley Eastern, 2002.
2. 2.John Chirillo and Scott Blaul : “Implementing Biometric Security”, 1st Edition, Wiley Eastern Publication, 2005.

REFERENCES:

John Berger: “Biometrics for Network Security”, 1st Edition, Prentice Hall, 2004.

For Theory courses only:

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

****

Course Code	Course Name	Course Category	LTP	Credits
20CS4263	**Human Computer Interaction**	**PEC**	**3-0-0**	**3**

Course Objective:

1. The Main Objective is the student to think constructively and analytically about how to design and evaluate interactive technologies.
2. Student can clearly understand the importance of the user interface design.
3. IT give an introduction to the key areas, approaches and developments in the field.
4. Basically, the course will introduce them to key areas, theoretical frameworks, approaches and major developments in HCI.
5. The main objective is to get student to think constructively and analytically about how to design and evaluate interactive technologies.

Course Content:

UNIT-I: Introduction and Graphical User Interface (9 Contact hours)

Importance of user Interface – definition, importance of good design. Benefits of good design. A brief history of Screen design.

The graphical user interface – popularity of graphics, the concept of direct manipulation, graphical system, Characteristics, Web user – Interface popularity, characteristics- Principles of user interface

UNIT-II: Design Process (7 Contact hours)

Human interaction with computers, importance of human characteristics human consideration, Human interaction speeds, understanding business junctions.

UNIT-III: Screen Designing (9 Contact hours)

Design goals – Screen planning and purpose, organizing screen elements, ordering of screen data and content – screen navigation and flow – Visually pleasing composition – amount of information – focus and emphasis – presentation information simply and meaningfully – information retrieval on web – statistical graphics – Technological consideration in interface design.

UNIT-IV: Windows (7 Contact hours)

New and Navigation schemes selection of window, selection of devices based and screen based controls.

UNIT-V: Components (7 Contact hours)

Text and messages, Icons and increases – Multimedia, colors, uses problems, choosing colors.

UNIT-VI: Software Tools and Interaction Devices (7 Contact hours)

Specification methods, interface – Building Tools; Keyboard and function keys – pointing devices – speech recognition digitization and generation – image and video displays – drivers.

Text Books:

1. Wilbert O Galitz,"*The essential guide to user interface design*", Wiley DreamaTech.
2. Ben Shneidermann ,"*Designing the user interface",* Pearson Education Asia 3rd Edition,.

References:

1. Alan Dix, Janet Fincay, Gre Goryd, Abowd, Russell Bealg *"Human – Computer Interaction",* Pearson, Third Edition.
2. Prece, Rogers, Sharps ,*"Interaction Design"* ,Wiley Dreamtech , Third Edition.
3. Soren Lauesen ,*"User Interface Design"*,Pearson Education.

Web resources:

1. Dr. Samit Bhattacharya , Dept. of Computer Science and Engineering, IIT Guwahati
   URL: http://nptel.ac.in/courses/106103115

Course outcomes: At the end of the course, the student will be able to

CO 1	Explain the capabilities of the both humans and computers from the view point of human Information processing
CO 2	Describe typical human-computer Interaction (HCI) models styles and various historic HCI paradigms.
CO 3	Apply an interactive design process and universal design principles to designing HCI system.
CO 4	Describe and use HCI design principles, standards and guidelines.
CO 5	Analyze and identify user models ,user support ,socio-organizational issues, and stakeholder requirements of HCI system.
CO 6	Design tasks and dialogs of relevant HCI system based on task analysis and dialog design.

Assessment Method

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

****

Course Code	Course Name	Course Category	L-T-P	Credits
20CS4264	Cyber Security	PEC	3-0-0	3

Course Learning Objectives:

1. To discuss basics of Information Security, related terminologies and the legal aspects.
2. To list assets and discuss the policies, standard and business continuity in an organization.
3. To illustrate risk assessment and explain the need of authentication and authorization.
4. To discuss firewall, IDS and various networking tools.
5. To explain the vulnerabilities present in the system, web and mobile environment.
6. To discuss the importance of physical security and the ways to maintain information security in any organization.

Course Content:

Unit - I **(8 Contact hours)**

History of Information Security, Understanding security, CNSS security model, Security in SDLC, Types of threats and attacks, Principles of Information Security, Laws and Ethics for Information Security, Introduction to IT ACT, International Laws and Legal bodies.

Unit - II **(8 Contact hours)**

Asset, Asset classification, Understanding the basics of Information Security Policy, Standards and Practices, Types of Policies, Policy development process, ISO 27001, Business Continuity Planning, Disaster Recovery, Maintaining Backups.

Unit - III **(8 Contact hours)**

Identification, Assessment, Analysis, Control of Risk, Quantitative vs Qualitative Risk Management, FAIR approach to risk assessment, NIST Risk management framework, Authentication vs Authorization, Types of authentication, Understanding different types of Access Controls(ACLs, RBAC, RUBAC etc.).

Unit - IV **(8 Contact hours)**

Overview of Firewalls, VPNs, DMZs, IDS, IPS, Honeypots, Detection methods, various network analysis tools.

Unit - V **(8 Contact hours)**

OS security, OWASP Top 10 web application security vulnerabilities, Mobile Devices risks, OWASP Top 10 mobile application security vulnerabilities.

Unit – VI **(5 Contact hours)**

Physical Vulnerability Assessment, Securing Assets, Physical Intrusion Detection, Procedures and Methods to maintain the implemented information Security.

Learning resources

Text book:

1. Whitman, Michael E., and Herbert J. Mattord, *'Principles of Information Security',* Cengage Learning India Private Limited, 5th edition

Reference Books:

1. Mark Rhodes and Ousley, *'Information Security: The Complete Reference',* McGraw-Hill Education; 2nd edition.
2. Stuart McClure , Joel Scambray and George Kurtz, *'Hacking Exposed 7'*, McGraw- Hill Education; 7th edition.

Course outcomes: At the end of the course, the student will be able to

CO 1	Analyze any organization for information security loopholes.
CO 2	Create and implement information security policies for any organization.
CO 3	Create and maintain information asset register.
CO 4	Implement Information Security framework for an organization.
CO 5	Identify the Information security risk, estimate its severity and recommend solutions.
CO 6	Test any application (system, web and mobile) for the technical risks.

Assessment Method:

Course Nature	**Theory**			
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

****

Course Code	Course Name	Course Category	L-T-P	Credits
20CS4265	**Robotics**	**PEC**	**3-0-0**	**3**

Course Learning Objectives:

students should be able to

1. To simulate the robot functions and joint movements.
2. Graphic animation sequences for robot movement.
3. Velocity, acceleration analysis of joint and linkages.
4. The basics and the latest technology of sensors used in robotics.
5. The different sensing variables
6. Robot vision system

Course Content:

UNIT I (Introduction to Robotics) (7 Contact hours)

What is robot and robotics, already designed robots, Manual and Autonomous robots, Different types of industrial ARM robots, and arm design, Coordinate transformations for more motor moments, Electrical connections of different boards and modules: How to connect closed circuit, digital and analog pins connections.

UNIT II (Robotic Principles) (7 Contact hours)

Robotic functions Geometrical commands, Edit commands. Selecting robot views, standard Robot part, using the parts in a simulation.

UNIT III (Robotic Motion) (7 Contact hours)

Types of motion, velocity and acceleration, Types of simulation motion Harmonic motion, parabolic motion, uniform motion velocity and acceleration analysis for robots.

UNIT IV (Actuators and sensors) (7 Contact hours)

Introduction to Arduino, Actuators, Sensors, Wired and wireless communication, I/O communication through USB cable, Bluetooth HC05, RF modules, DTMF module, Xbee modules.

UNIT V (Robotics Simulation) (8 Contact hours) Simulation packages, Loading the simulation, Simulation editors, delay, Resume commands. Slide commands, program flow control. Robot motion control, Analysis of robot elements, Robotic linkages.

UNIT VI (Basic robots and Raspberry pi) (10 Contact hours)

Line follower: Line follower robot design and control with Arduino board, Obstacles avoider: Obstacle avoider robot with IR sensors and Arduino board, Mobile controller: Mobile controller robot with DTMF module and HC05 module, Introduction to Raspberry pi: What is raspberry and differences between Arduino and raspberry pi, Applications of robotics.

Learning resources:

Text book:

1. Robotics for engineers by Yoram Koren
2. John J. Craig - *Introduction to Robotics: Mechanics and Control, Pearson, Upper Saddle River*, NJ, 2005.
3. Marco Schwartz - *Internet of Things with Arduino Cookbook*.
4. P.A. Janaki Raman, *Robotics and Image Processing an Introduction*, Tata Mc Graw Hill Publishing company Ltd., 1995.
5. Richard D. Klafter, Thomas .A, Chri Elewski, Michael Negin, *Robotics Engineering an Integrated Approach*, Phi Learning., 2009.
6. Mikell P Groover & Nicholas G Odrey, Mitchel Weiss, Roger N Nagel, Ashish Dutta, *Industrial Robotics, Technology programming and Application*s, Tata McGraw-Hill Education, 2011.

Reference Books:

1. Myke Predko, "*Programming Robot Controllers"* – McGrawHill, 1st edition, 2003.
2. Murphy Robin R*," Introduction to AI Robotics*", MIT Press, 2000.
3. Siegwart R and Nourbakhsh I.R, "*Introduction to Autonomous mobile Robots*", Prentice Hall India, 2005.

Web resources:

https://ieeexplore.ieee.org/document/833572/ by K Taylor - 2000
https://www.iste.org/explore/articleDetail?articleid=846&category=In-the.Nov 1, 2016
www.ieee-ras.org/...resources.../educational-material-in-robotics-and-automation2015 - SyRoTek - System for **robotic** e-learning

Course outcomes: At the end of the course, the student will be able to

CO 1	knowledge of programming and algorithms, as well as systems development techniques
CO 2	knowledge of digital technology that forms the basis for designing computers and embedded systems.
CO 3	knowledge about electronics and sensors that form the basis for machine launching.
CO 4	knowledge of mathematics required for electronics, programming and robotics.
CO 5	knowledge about the management and analysis of robotics systems.
CO 6	knowledge of how machines can interpret surroundings, act intelligently and adapt.

For Theory courses only:

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total

Weightage (%)	10%	30%	60%	100%

****

Course Code	Course Name	Course Category	L-T-P	Credits
20CS4266	**Computer Forensics**	**PEC**	**3-0-0**	**3**

Course Learning Objectives:

1. To discuss basics of computer forensics, types and role of data backup.
2. To illustrate the procedure of evidence collection and data seizure.
3. To inspect network, identify crime and review a case.
4. To choose appropriate forensics tool to perform investigation.
5. To examine mobile phones and inspect windows and Dos for evidence gathering.
6. To illustrate the guidelines for writing reports and responsibilities of expert witness.

Course Content:

Unit - I (8 Contact hours)

Computer Forensics Fundamentals: What is Computer Forensics?, Use of Computer Forensics in Law Enforcement, Computer Forensics Assistance to Human Resources/Employment Proceedings, Computer Forensics Services, Benefits of Professional Forensics Methodology, Steps taken by Computer Forensics Specialists Types of Computer Forensics Technology: Types of Military Computer Forensic Technology, Types of Law Enforcement, Computer Forensic Technology, Types of Business Computer Forensic Technology Computer Forensics Evidence and Capture: Data Recovery Defined, Data Back-up and Recovery, The Role of Back-up in Data Recovery, The Data-Recovery Solution.

Unit - II (8 Contact hours)

Evidence Collection and Data Seizure: Why Collect Evidence? Collection Options, Obstacles, Types of Evidence, The Rules of Evidence, Volatile Evidence, General Procedure, Collection and Archiving, Methods of Collection, Artifacts, Collection Steps, Controlling Contamination: The Chain of Custody Duplication and Preservation of Digital Evidence: Preserving the Digital Crime Scene, Computer Evidence Processing Steps, Legal Aspects of Collecting and Preserving Computer

Forensic Evidence Computer Image Verification and Authentication: Special Needs of Evidential Authentication, Practical Consideration, Practical Implementation.

Unit - III **(8 Contact hours)**

Computer Forensics analysis and validation: Determining what data to collect and analyze, validating forensic data, addressing data-hiding techniques, performing remote acquisitions.

Network Forensics: Network forensics overview, performing live acquisitions, developing standard procedures for network forensics, using network tools, examining the honeynet project.

Processing Crime and Incident Scenes: Identifying digital evidence, collecting evidence in private-sector incident scenes, processing law enforcement crime scenes, preparing for a search, securing a computer incident or crime scene, seizing digital evidence at the scene, storing digital evidence, obtaining a digital hash, reviewing a case.

Unit - IV **(8 Contact hours)**

Current Computer Forensic tools: evaluating computer forensic tool needs, computer forensics software tools, computer forensics hardware tools, validating and testing forensics software E-Mail Investigations: Exploring the role of e-mail in investigation, exploring the roles of the client and server in e-mail, investigating e-mail crimes and violations, understanding e-mail servers, using specialized e-mail forensic tools.

Unit - V **(8 Contact hours)**

Cell phone and mobile device forensics: Understanding mobile device forensics, understanding acquisition procedures for cell phones and mobile devices.

Working with Windows and DOS Systems: understanding file systems, exploring Microsoft File Structures, Examining NTFS disks, Understanding whole disk encryption, windows registry, Microsoft startup tasks, MS-DOS startup tasks, virtual machines.

Unit – VI **(5 Contact hours)**

Report writing and Expert witness: Importance of reports, guidelines to write reports, using tools to generate reports, guidelines for expert witness and testifying in court, ethical responsibilities in expert testimony.

Learning resources

Text book:

1. Amelia Phillips and Bill Nelson, *'Guide to Computer Forensics and Investigations',* Cengage Learning India Private Limited, 4th edition.

Reference Books:

1. John R. Vacca, *'Computer Forensics: Computer Crime Scene Investigation',* Charles River Media, 2nd edition.
2. Eoghan Casey, *'Handbook of Digital Forensics and Investigation',*Academic Press, 1st edition.

Course outcomes: At the end of the course, the student will be able to

CO 1	Examine any device and recover data from it.
CO 2	Analyze any device for collecting evidence and helping the law officials in prosecuting a person for any crime committed.
CO 3	Demonstrate the use of various forensic tools for conducting investigation.

For Theory courses only:

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

****

OPEN ELECTIVE COURSES FOR ALL BRANCHES Except CSE

Course code	Course Name	Course Category	L-T-P	Credits
20CSXX71	**Object Oriented Programming through JAVA**	**PCC**	**3-0-0**	**3**

Course Learning Objectives:

1. Gain knowledge about basic Java language syntax and semantics to write Java programs and use concepts such as variables, conditional and iterative execution methods etc.,
2. Understanding the fundamentals of object-oriented programming in Java, including defining classes, objects, invoking methods etc. and exception handling mechanisms.
3. Understand the principles of inheritance, packages and interfaces.
4. Understand the principles of Multithreading and Event handling mechanisms.

Course Content:

Unit 1:

Introduction: Object Oriented Programming, Introduction to java and JVM, Key features, Fundamentals of Objects and Classes, Access Specifiers, data types, dynamic initialization, scope and life time, operators, Conditional Statements, control structures, arrays, type conversion and casting. Constructors, usage of static, access control, this key word, garbage collection, overloading, parameter passing mechanisms, nested classes and inner classes.

Unit II:

Strings: Exploring the String class, String buffer class, Command-line arguments. Library: StringTokenizer, Random class, Wrapper classes. Encapsulation: Abstraction. Creating User defined Data Structures: Array of Objects, User defined Linked List.

Unit III:

Inheritance and Interface:Types of Inheritance, usage of super key word, method overriding, final methods and classes, abstract classes, Polymorphism: dynamic method dispatch, Static method dispatch. **Interfaces**: Differences between classes and interfaces, defining an interface, implementing interface, variables in interface and extending interfaces.

Unit IV:

File Handling: Streams, File class, File streams. File Reader, File Writer, Buffered Reader, Buffered Writer, String Tokenizer **Exception Handling:** Concepts of Exception handling, types of exceptions, usage of try, catch, throw, throws and finally keywords, Built-in exceptions, creating own exception sub classes.

Unit V

Packages: Creating a Package, setting CLASSPATH, Access control protection, importing packages.**Multithreading :** Concepts of Multithreading, differences between process and thread, thread life cycle, Thread class, Runnable interface, creating multiple threads, Synchronization, thread priorities, inter thread communication, daemon threads, deadlocks, thread groups.

Unit VI (8Contact hours)

Event Handling: Introduction to Event Handling, AWT Components, windows, Layout Managers, Event handling model of AWT, Adapter classes, Menu, Menu bar. **Swings**: swings introduction, JFrame, JPanel and JComponent, Icons and Labels, text fields, buttons – The JButton class, Check boxes, Radio buttons. Combo boxes, Action Listeners. Introduction to JDBC.

Learning Resources

Text books:

- Herbert Schildt, "The Complete Reference Java", TMH Publishing Company Ltd, 9th Edition.
- Cay Horstmann, "Big Java", John Wiley and Sons, 2nd Edition

Reference Books:

- Allen B.Downey,"Think Java; How to Think Like a Computer Scientist",Paper Back 1stEdition
- David J. Eck, Hobart and William Smith Colleges, "Introduction to Programming Using Java" Published by Paper Back .

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

Web resources:

1. http://www.nptelvideos.com/java/java_video_lectures_tutorials.php
2. https://www.tutorialspoint.com/java/
3. https://www.javatpoint.com/java-tutorial
4. http://mooc.fi/courses/2013/programming-part-1/material.html
5. http://math.hws.edu/javanotes

Course Outcomes: At the end of the course, the student will be able to

CO 1	Explain OOP Principles and Write Basic Java Programs.
CO 2	Defining Classes and Objects. Identify classes, objects, members of a class and relationships among them needed for a specific problem
CO 3	To be able to write Java Programs to demonstrate method overloading and Demonstrate the concepts of polymorphism and inheritance. Discuss method overriding V/s method overloading.
CO 4	Explain the benefits of JAVA's Exceptional handling mechanism compared to other Programming Language
CO 5	To be able to write Java Programs to demonstrate Packages and Threading concepts.
CO 6	Discuss and Demonstrate the AWT Concepts and develop the AWT Applications.

For Theory courses only:

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

****

Course code	**Course name**	**Course Category**	**L-T-P**	**Credits**

20CSXX72	Database Management Systems	PCC	3-0-0	3

Course Learning Objectives:

1. To Understand the role of a database management system in an organization.
2. To Understand the basics of ER Diagram, Relational model, Relational Algebra and Relational Calculus.
3. To Understand basic database concepts, including the structure and operation of the relational data model.
4. To Construct simple and moderately advanced database queries using Structured Query Language (SQL).
5. To Understand and successfully apply logical database design principles, including E-R diagrams and database normalization.
6. To Understand the concept of a database transaction and related database facilities and indexing techniques.

Course Content:

Unit I (8Contacts hours)

Introduction to database systems, File System vs. Database Systems, Database system structure, Views of data in a database system, Data models and Database languages.

Introduction to Entity-Relationship data model, Elements of an ER model, Constructing ER diagrams, Modelling of constraints, Reduction of ER diagrams to relational tables.

Unit II (6 Contact hours)

Basics of relational model, ER diagrams to relational design, Relational algebra: Simple operations and extended operations, writing relational algebra expressions for queries, Introduction to tuple relational calculus and writing basic queries using tuple calculus

Unit III (9 Contact Hours)

Basic structure of SQL queries, Writing simple queries, Complex queries and nested Subqueries in SQL, Aggregate functions in SQL, Effect of NULL values on result, Defining a Relational Schema, View definitions and constraints, types of keys.

Unit IV (7 Contact hours)

Features of Good Relational Designs, Atomic Domains and First Normal Form, Problems encountered in bad schema design, Motivation for normal forms, Dependency theory-functional dependencies, Armstrong's Axioms for FD, Closure of a set of FD's, Minimal Cover, Definition of 1NF, 2NF, 3NF and BCNF, Decomposition and desirable properties of them.

Unit V (6Contact hours)

Storing data in disk and files and the memory hierarchy, RAID, File organization and indexes, ISAM Tree, B+ Tree, Linear Hashing and Extendible Hashing

Unit VI (9 Hours)

Transaction concept, ACID properties, Concurrency in a DBMS, Serializability and Recoverability, Concurrency control Protocols (lock-based and time-stamp based)

Text Books

A. Silberschatatz, H. F. Korth and S. Sudarshan, Database System Concepts, 5/e, McGraw Hill,2006

R. Ramakrishnan and J. Gehrke, Database System Concepts, 3/e, McGraw Hill, 2003

Wilfried Lemahieu, Seppe Vanden Broucke and Bart baesens Principles of Database Management Systems, 1/e Cambridge 2018

Reference Books

Ramez Elmasri, Shamkant B. Navathe , Fundamentals of Database (7th Edition), Paperback, 2007

Theorey T J, Database Modeling & Design, 2/e, Morgan Kaufmann Publishers, 1994.

H. GarciaMolina, J. D. Ullman and J. Widom, Database Systems The Complete Book, 1/e, Pearson Education, 2007

Web resources:

Department of CS&E, IIT M, "Introduction to Database Sytems and Design", https://nptel.ac.in/courses/106106095/

Indian Institute of Technology, Kharagpur, " Database Management Systems", https://nptel.ac.in/courses/106105175/

Course outcomes: At the end of the course, the student will be able to

CO 1	Demonstrate the basic elements of a relational database management system,
CO 2	Ability to identify the data models for relevant problems.
CO 3	Ability to design entity relationship and convert entity relationship diagrams into RDBMS and formulate SQL queries on the respect data.
CO 4	Apply normalization for the development of application software
CO 5	Ability to learn about Disk Management, Buffer management
CO 6	Ability learn about transaction management

For Theory courses only:

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

Course Code	CourseNname	Course category	L-T-P	Credits
20CSXX73	**COMPUTER GRAPHICS**	**PEC**	**3-0-0**	**3**

OBJECTIVES:

1. Gain knowledge about graphics hardware devices and software used. · Understand the two dimensional graphics and their transformations.
2. Understand the three dimensional graphics and their transformations. · Appreciate illumination and color models.
3. Be familiar with understand clipping techniques.

Course Content:

UNIT I INTRODUCTION (8 Contact hours)

Survey of computer graphics, Overview of graphics systems – Video display devices, Raster scan systems, Random scan systems, Graphics monitors and Workstations, Input devices, Hard copy Devices, Graphics Software; Output primitives – points and lines, line drawing algorithms, loading the frame buffer, line function; circle and ellipse generating algorithms; Pixel addressing and object geometry, filled area primitives.

UNIT II Two Dimensional Graphics (8 Contact hours)

Two dimensional geometric transformations – Matrix representations and homogeneous coordinates, composite transformations; Two dimensional viewing – viewing pipeline, viewing coordinate reference frame; widow-to-viewport coordinate transformation, Two dimensional viewing functions; clipping operations – point, line, and polygon clipping algorithms.

UNIT III Three Dimensional Graphics (8 Contact hours)

Three dimensional concepts; Three dimensional object representations – Polygon surfaces- Polygon tables- Plane equations – Polygon meshes; Curved Lines and surfaces, Quadratic surfaces; Blobby objects; Spline representations – Bezier curves and surfaces -B-Spline curves and surfaces. TRANSFORMATION AND VIEWING: Three dimensional geometric and modeling transformations – Translation, Rotation, Scaling, composite transformations; Three dimensional viewing – viewing pipeline, viewing coordinates, Projections, Clipping; Visible surface detection methods.

UNIT IV Illumination And Colour Models (7 Contact hours)

Light sources – basic illumination models – halftone patterns and dithering techniques; Properties of light – Standard primaries and chromaticity diagram; Intuitive colour concepts – RGB colour model – YIQ colour model – CMY colour model – HSV colour model – HLS colour model; Colour selection.

UNIT V Animations & Realism 10 Animation Graphics (8 Contact hours)

Design of Animation sequences – animation function – raster animation – key frame systems – motion specification –morphing – tweening. COMPUTER GRAPHICS REALISM: Tiling the plane – Recursively defined curves – Koch curves – C curves – Dragons – space filling curves – fractals – Grammar based models – fractals – turtle graphics – ray tracing.

UNIT –VI Multimedia System: (8 Contact hours)

An Introduction, Multimedia hardware, Multimedia System Architecture. Data & File Format standards. i.e RTF, TIFF, MIDI, JPEG, DIB, MPEG,Audio: digital audio, MIDI, processing sound, sampling, compression. Video: Avi, 3GP,MOV, MPEG .

Text Books:

1. John F. Hughes, Andries Van Dam, Morgan Mc Guire ,David F. Sklar , James D. Foley, Steven K. Feiner and Kurt Akeley ,"Computer Graphics: Principles and Practice", , 3rd Edition, Addison- Wesley Professional,2013. (UNIT I, II, III, IV).
2. Donald Hearn and Pauline Baker M, "Computer Graphics", Prentice Hall, New Delhi, 2007 (UNIT V).

References:

1. Donald Hearn and M. Pauline Baker, Warren Carithers,"Computer Graphics With Open GL", 4th Edition, Pearson Education, 2010.
2. Jeffrey McConnell, "Computer Graphics: Theory into Practice", Jones and Bartlett Publishers, 2006.
3. Hill F S Jr., "Computer Graphics", Maxwell Macmillan" , 1990.
4. Peter Shirley, Michael Ashikhmin, Michael Gleicher, Stephen R Marschner, Erik Reinhard, KelvinSung, and AK Peters, Fundamental of Computer Graphics, CRC Press, 2010.
5. William M. Newman and Robert F.Sproull, "Principles of Interactive Computer Graphics", Mc GrawHill 1978. http://nptel.ac.in/

Outcomes:

CO1	Design two dimensional graphics.
CO2	Apply two dimensional transformations.
CO3	Design three dimensional graphics.
CO4	Apply three dimensional transformations.
CO5	Apply Illumination and color models.
CO6	Apply clipping techniques to graphics.
CO7	Design animation sequences.

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total

Weightage (%)	10%	30%	60%	100%

****

Course Code	Course Name	Course Category	L-T-P	Credits
20CSXX74	**Distributed Computing**	**PEC**	**3-0-0**	**3**

Course Learning Objectives:

1. To expose students to both the abstraction and details of file systems
2. To introduce concepts related to distributed computing systems.
3. To focus on performance and flexibility issues related to systems design decisions.
4. To prepare students for life-long learning
5. To evaluate how and not just the memorize the details.
6. To expose students to current literature in distributed systems.

Course Content:

UNIT – I **(8 Contact hours)**

Fundamentals: Evolution of Distributed Computing Systems, System models, issues in design of Distributed Systems, Distributed computing environment, web based distributed model, computer networks related to distributed systems and web based protocols

UNIT-II **(8 Contact hours)**

Message Passing: Inter process Communication, Desirable Features of Good Message-Passing Systems, Issues in IPC by Message, Synchronization, Buffering, Multidatagram Messages, Encoding and Decoding of Message Data, Process Addressing, Failure Handling, Group Communication

UNIT-III **(7 Contact hours)**

Remote Procedure Calls: The RPC Model, Transparency of RPC, Implementing RPC Mechanism, Stub Generation, RPC Messages, Server Management, Communication Protocols for RPCs, Complicated RPCs, Client-Server Binding, Exception Handling,Some Special Types of RPCs, Lightweight RPC, Optimization for Better Performance.

UNIT- IV **(8 Contact hours)**

Distributed Shared Memory: Design and Implementation issues of DSM, Granularity, Structure of Shared memory Space, Consistency Models, replacement Strategy, Thrashing, Other Approaches to DSM, Advantages of DSM.

UNIT – V **(8 Contact hours)**

Synchronization: Clock Synchronization, Event Ordering, Mutual Exclusion, Election Algorithms. Constant.

Resource and Process Management: Desirable Features of a good global scheduling algorithm, Load Balancing approach, Load Sharing Approach, Process Migration, Threads, Processor allocation, Real time distributed Systems.

UNIT – VI **(8 Contact hours)**

Distributed File Systems: Desirable Features of a good Distributed File Systems, File-sharing Semantics, File Models, File Accessing Models,ile caching Schemes, File Replication, Design Principles, Sun's network file system, Andrews file system, comparison of NFS and AFS.

Learning resources

Text book:

1. Ajay D. Kshemkalyani, Mukesh Singhal , "Distributed Computing: Principles, Algorithms, and Systems Reissue Edition",
2. Sukumar Ghosh, Distributed Systems: An Algorithmic Approach, Second Edition

Reference Books:

1. Distributed OS by Pradeep K. Sinha (PHI)
2. Tanenbaum S.: Distributed Operating Systems, Pearson Education
3. Tanenbaum S. Maarten V.S.: Distributed Systems Principles and Paradigms, (Pearson Education)
4. George Coulouris, Jean Dollimore. Tim Kindberg: Distributed Systems concepts and design.

Web resources:

https://eclass.uoa.gr/modules/document/file.php/D245/2015/DistrComp.pdf

Course outcomes: At the end of the course, the student will be able to

CO 1	Study software components of distributed computing systems.
CO 2	Know about the communication and interconnection architecture of multiple computer systems.
CO 3	Recognize the inherent difficulties that arise due to distributed-ness of computing resources.
CO 4	Understanding of networks & protocols, mobile & wireless computing and their applications to real world problems
CO 5	Able to use pointers in C programming

For Theory courses only:

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

****

Course Code	Course Name	Course Category	L-T-P	Credits
20CSXX75	**Digital Image Processing**	**PEC**	**3-0-0**	**3**

Course Objectives:

1. To describe the image fundamentals and mathematical transforms necessary for image processing.
2. To explain the image enhancement techniques
3. To analyze images in the frequency domain using various transforms.
4. To explain image restoration procedures.
5. To describe Image Compression and Segmentation used in digital image processing.
6. To describe image feature extraction methods.

Course Outcomes:

1. Be able to apply, design and implement solutions for digital image processing problems.
2. Be able to discuss the strengths and limitations of DIP applications in solving problems with both professional peers and lay clients.

Course content:

UNIT I **(8 Content hours)**

Digital image fundamentals – Electromagnetic spectrum and imaging, Image acquisition, image formation. Digitization-sampling and quantization, Resolution-pixel, gray scale, spatial, basic relationship between pixels, Distance measure, Mathematical operations on image, Geometrical and spatial transformation.

UNIT II **(8 Content hours)**

Intensity transformation and spatial filtering: Image enhancement, log transformation, Gamma transformation, Histogram processing, Histogram matching, Special filtering- spatial correlation and convolution, generating spatial filter mask, mage smoothing, Image sharpening-Laplacian filter, Highboost filter. Edge detection- gradient filter, Morphological image processing-erosion, Dilation, opening and closing operations, Boundary extraction, Hole Filling, Extraction of connected components, Thinning, and thickening.

UNIT III **(7 Content hours)**

Image Restoration-Noise model, Restoration-Mean filter, Geometric filter, median filter, adaptive filter, band pass filter, Notch filter, least mean square filters. Color fundamental-RGB color model, CMY color model, HSI color model. Converting RGB to HSI and vice-versa.

UNIT IV **(7 Content hours)**

Filtering in Frequency domain-Preliminary concept: Fourier series, Fourier transform, convolution, Sampling, DFT, Enhancement in frequency domain, low pass filter, high pass filter. Computing IDFT from DFT.

UNIT V **(8 Content hours)**

Image compression fundamental, coding, temporal and spatial redundancy, Error-free (Lossless) and Lossy compression. Image segmentation, Point-line-edge detection. Image gradients operator, canny edge detection, Edge linking and boundary detection, local processing, thresholding, variable thresholding, Region Growing, Texture Segmentation; Region oriented segmentation.

UNIT VI **(8 Content hours)**

Feature Extraction: Edges – Canny, Sobel; Line detectors, Corners - Harris, Orientation Histogram, SIFT, SURF, Scale-Space Analysis- Image Pyramids, Haar transform. Decision-theoretic and structure descriptors.

Textbooks:

1. Digital Image processing – R.C. Gonzalez & R.E. Woods, Addison Wesley/ Pearson education, 3rd edition.
2. D. A. Forsyth, J. Ponce, Computer Vision: A Modern Approach, PHI Learning 2009.
3. Milan Soanka, Vaclav Hlavac and Roger Boyle, Digital Image Processing and Computer Vision, Cengage Learning.

REFERENCES:

Fundamentals of Digital Image processing – A.K.Jain , PHI

For Theory courses only:

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

****

COURSE OFFEED TO OTHR ENGINEERING BRANCHED BY DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Course Code	Course Name	Course Category	L-T-P	Credits
20CSXX08 (All branches except CSE)	**Programming and Data Structures**	**ESC**	**3: 0: 0**	**3**

Course Learning Objectives

1. To deduce adequate knowledge in programming language and problem-solving techniques.
2. To develop programming skills using the fundamentals of CLanguage.
3. To recognize the effective usage of arrays, structures, functions,pointers.
4. To implement the memory managementconcepts.
5. To illustrate the usage of pointers and dynamic memoryallocation.
6. Explore Data Structures and itsapplications.

Course Content

Unit-I **(5hours)**

Introduction

Computer

Hardware, Bits and Bytes, History of Programming Languages, Character Set, Variables and Identifiers, Built-in Data Types. Operators and Expressions, Constants and Literals, Simple Assignment Statement, Basic Input/output Statement, Simple 'C' Program, Conditional Statements andLoops.

Unit – II *(6 hours)*

Arrays

One Dimensional Arrays, Array Manipulation, Searching, Insertion, Deletion of An Element from An Array; Finding the Largest/Smallest Element in An Array; Two

Dimensional Arrays, Addition/Multiplication of Two Matrices, Transpose of square Matrix, Inverse of Matrix, Character Arrays, Multi-dimensionalarrays.

Unit ¯ III *(8hours)*

Functions

Function Declaration, Function Definition, Function Call, Call by Value, Call byReference, Recursion, String Fundamentals, String HandlingFunctions.

Unit -IV *(8 hours)*

Structure & Union

Structure Variables, Initialization, Structure Assignment, Nested Structure, Structures and Functions, Structures and Arrays: Arrays of Structures, Structures Containing Arrays, Unions.

Unit -V *(8hours)*

Pointers

Pointer Type Declaration, Pointer Assignment, Pointer Initialization, Pointer Arithmetic, Functions and Pointers, Arrays and Pointers, Pointer to Pointers, Dangling Memory, Dynamic Memory Allocations, Storage Classes.

Unit VI *(10hours)*

Data Structures

Linked List, Double Linked Lists, Stack, Stack Implementation Using Arrays, Stack Implementation Using Linked List, Queues, tree traversals.

Learning Resources Text book

1. ReemaThareja, *'Data Structures using C'*, Oxford Higher Education,2nd Edition.

Reference Books

1. W. Kernighan, DennisM. Ritchie, *'C ProgrammingLanguage'*, Prentice Hall India Learning Private Limited, 2nd Edition.
2. Balagurusamy, *'Programming in C'*, McGraw Hill Education India Private Limited; 7th Edition.
3. YashavantKanetkar, *'Let us C'*, BPB Publications,14th Edition

Web resources

1. Prof Satyadev Nandakumar, NPTEL-IIT Kanpur, *'Introduction to Programming in C'*, URL:https://nptel.ac.in/syllabus/106104128/
2. Dr P P Chakraborty, NPTEL-IIT Kharagpur, *'Programming and DataStructures'* URL: https://nptel.ac.in/courses/106105085/4
3. URL:https://www.tutorialspoint.com/cprogramming/

Course outcomes: At the end of the course, the student will be able to

Rajiv Gandhi University of Knowledge Technologies - AP

Department of Electronics & Communications Engineering

CO 1	Illustrate the flowchart and design an algorithm for a given problem and to develop one C program using Operators.
CO 2	Develop conditional and iterative statements to write C Programs.
CO 3	Describe C Programs that use the arrays and its usage.
CO 4	Exercise user defined functions to solve real time problems.
CO 5	Describe C Programs using pointers and to allocate memory using dynamic memory management functions.

CO 6	Explore different data structures and understand.

Assessment Method

Assessment Tool	Weekly tests/Assignments (in semester)	Monthly tests (in semester)	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

****

Course code	**Course name**	**Course type**	**L: T: P**	**credits**
20CSXX88 **(All branches except CSE)**	**Programming and Data Structures Laboratory**	**ESC**	**0: 0: 3**	**1.5**

Course Learning Objective

1. Understand the basic concept of C Programming and Data Structures, its different modules that include conditional and looping expressions, Arrays, Strings, Functions, Structures, Files, Stacks andQueues.
2. Acquire knowledge about the basic concept of writing aprogram.
3. Purpose of programming language and its application in problemsolving.

List of Experiments

Exercise-1: Introduction to C, Conditional Statements and Loops

1. C Program to calculate the sum of Naturalnumbers.
2. C Program to generate multiplication table of a givennumber.

3. C Program to display Fibonacci sequence (Up to givennumber).
4. C Program to Check whether a given number is prime ornot.
5. C Program to make a simple Calculator using switchcase.
6. C Program to check whether a number is palindrome ornot.
7. C Program to display factors of a givennumber.
8. C Program to print Pyramids, Triangles and various patters usingloops.

Exercise-2: Arrays and Sorting

1. C Program to find second largest Element of anArray.
2. C Program to add two matrix using multi-dimensionalarrays.
3. C Program to multiply two matrix using multi-dimensionalarrays.
4. C Program to find transpose of amatrix.
5. C Program to Sort Elements of an Array using Bubblesort.
6. Using Insertion Sort, SelectionSort.
7. Using Counting Sort, Bucket Sort 8. Check whether two strings are anagram of each other ornot.

Exercise 3: Functions and Recursion

1. C Program to check whether given number is prime or not using user-defined function.
2. C Program to swap two integer values using call by value and call byreference.
3. C Program to find the factorial of a given number usingrecursion.

4. C Program to calculate length of string without using strlen()function.
5. C Program to print all permutations of a string (abc, acb, bac, bca, cab,cba).
6. C Program to sort elements in Lexicographical order (Dictionary order) using in built stringfunctions.
7. Sorting using MergeSort.
8. Sorting using QuickSort.

Exercise-4: Structues and Unions

1. C Program using structures to read and display the information about astudent.
2. C Program to read, display, add and subtract two complexnumbers.
3. C Program to read and display the information of a student using nestedstructure
4. C Program, using an array of pointers to a structure, to read and display the data of students.
5. C Program to demonstrate arrays of Unionvariables.
6. C Program using structures to maintain a book library (Book is a structure) which has following operations print various types of books along with their count, author details, search a book by author name or book name orpublisher.

Exercise-5: Pointers and File Handling

1. C Program to demonstrate, handling of pointers inC.
2. C Program to access array elements usingpointers.
3. C Program to find the sum of n numbers with arrays andpointers.
4. C Program to swap two numbers using pointers andfunction
5. C Program to find sum of n elements entered by user. To perform this allocate memory dynamically using malloc()function.
6. C Program to read and write afile.
7. C Program to count number of lines andwords.
8. Write a c program to copy a data of file to otherfile.

Exercise-6: Introduction to Data Structures

1. Write a program to create a linked list and perform insertions and deletions of all cases. Write functions to sort and finally delete the entire list atonce.
2. Write a program to create a doubly linked list and perform insertions and deletions in allcases.
3. Write a program to perform push, pop and peek operations on astack.
4. Write a program to implement a linkedstack.
5. Write a program to implement a linkedqueue.
6. Write a program to implement binary search treeinsertion.

7. Write a program to implement binary search tree traversals (pre-order, post-order, in-order).

Course outcome: After the completion of this Laboratory course, the student will be able to

CO 1–	Apply and practice logical ability to solve the problems
CO 2	Understand C programming development environment, compiling, debugging, executing a program using the development environment
CO 3	Analyzing the complexity of problems, modularize the problems into small modules and then convert them into programs
CO 4	Understand and apply the in-built functions and customized functions for solving the problems
CO 5	Understand and apply the pointers, memory allocation techniques and use of files for dealing with variety of problems
CO 6	Understand and apply the structures and unions concept and solving problems on the same
CO 7	Understand the basic concepts of stacks, queues and applying the same for basic problems

Assessment Method

Assessment Tool	Experiments	Record	Viva-Voce/ Quiz/MCQ/Lab project	Total
Weightage (%)	25%	5%	10%	40%
End Semester Examination weightage (%)				60%

****

Course Code	Course Name	Course Category	L-T-P	Credits
20CSXX09 (ECE)	**Object Oriented Programming**	**ESC**	**2- 0- 0**	**2**

Rajiv Gandhi University of Knowledge Technologies - AP

Department of Electronics & Communications Engineering

Course Learning Objectives

1. Gain knowledge about basic C++ language syntax and semantics to write C++ programs and use concepts such as variables, conditional and iterative execution methods etc.,
2. Understanding the fundamentals of object-oriented programming inC++, including defining classes, objects, invoking methods etc. and exception handling mechanisms.
3. Understand the principles of inheritance, packages andinterfaces.
4. Understand the principles of Multithreading and Appletprogramming

Course content

Unit-1: Review of C: strings, arrays, pointers, Programming in C++ : Build and execute a C program in C++, Write equivalent programs in C++, C++ as Better C : Procedural Extensions of C

Unit-2: OOP in C++: Classes and basic Object-Oriented features (encapsulation), Overview of OOP in C++: More OO features, overloading, namespace and using struct and union

Unit-3: Inheritance : Generalization / Specialization of Object Modeling in C++, Polymorphism : Static and Dynamic Binding.

Unit-4: Type Casting &Exceptions : C++ cast operators; C++ Exceptions & standard exception

Unit-5: Classes Templates& STL - Function and Class templates and using STL like containers, algorithms.

Unit-6: File handling, streams, Interfaces and Multithreaded Programming.

References:

1. C++ Primer, Stanley Lippman, 5th edition.
2. Object-Oriented Programming with C++, E.Balagurusamy, McGraw-Hill Education (India)

Web resources:

1. PROF. PARTHA PRATIM DAS, IIT Kharagpur, NPTEL," PROGRAMMING IN C++" NPTEL :: Computer Science and Engineering - NOC:Programming in C++
1. Object Oriented Programming in C++ - GeeksforGeeks

****

Course Code	Course Name	Course Category	L-T-P	Credits

Rajiv Gandhi University of Knowledge Technologies - AP

Department of Electronics & Communications Engineering

20CSXX89 (ECE)	**Object Oriented Programming Laboratory**	**ESC**	**0: 0: 3**	**1.5**

Course Learning Objective

1. To build software development skills using C++ programming for real-world applications.
2. To understand and apply the concepts of classes, packages, interfaces, arraylist, User defined Linked List, File Handling, exception handling andMulti-threading.

List of Experiments

Lab No 1: Basic Programs in C++.

Lab No 2: Programming Assignments on Arrays and Strings.

Lab No 3: Programming Assignments on Classes, Objects and Encapsulation. Lab No 4: Implementing the concepts of Inheritance and Array Objects.

Lab No 5: Implementing the OOPS Concepts of Abstract, Interfaces and Polymorphism. Lab No 6: Programming Assignments on File Handling.

Lab No 7: Programming Exercises on Exception Handling. Lab No 8: Working with List Operations.

Lab No 9: Implementing the concepts of Multi-Threading.

Course Outcomes

At the end of the course, the student will be able to

CO 1	Understanding the control structures and conditional statements in C++
CO 2	Understanding the arrays and String handling in C++
CO 3	Understanding the difference between class and object and providing security for objects
CO 4	Understanding the reusability of objects and working with multiple objects

CO 5	Understanding about hiding the data, getting multiple inheritance through Interfaces
CO 6	Understanding the data processing from files
CO 7	Understanding about handling run time abnormal program executions
CO 8	Understanding about creating user defined linked list and dynamic objects
CO 9	Understanding the multi-threaded programming and inter thread Communication

Rajiv Gandhi University of Knowledge Technologies - AP

Department of Electronics & Communications Engineering

Assessment Method

Assessment Tool	Experiments	Report/Viva-Voce/ Quiz/MCQ/Lab Project	Total
Weightage (%)	25%	15%	40%
End Semester Examination weightage (%)			60%

****

Course Code	Course Name	Course Category	L-T-P	Credits
20CSXX10 (ECE)	**Computer Organization and Architecture**	**ESC**	**3: 1: 0**	**4**

Course Learning Objectives:

To expose the students to the following:

1. How Computer Systems work & the basicprinciples.
2. Instruction Level Architecture and InstructionExecution.
3. The current state of art in memory systemdesign.
4. How I/O devices are accessed and itsprinciples.
5. To impart the knowledge on microprogramming.

Course Content

Unit – I **(10 hours)**

Architecture of 8086 microprocessor, special functions of general purpose registers,8086 flag register and function of 8086 flags, pin diagram of 8086,minimum and maximum mode of 8086 configuration and timing diagrams.Addressing modes of 8086, Instruction sets of 8086.

Unit II ***(12 hours)***

Introduction to MIPS architecture, MIPS Instruction Set Architecture, Procedures, Recursive Programs, Architecture Examples, Introduction to Assessing and Understanding Performance, CPU Performance and its Factors, Evaluating Performance, Benchmarks and the performance of recent IntelProcessors.

Unit III ***(12hours)***

Introduction to Processor: Data path and Control, Logic design Conventions, Building a Data path, Simple Implementation scheme, Multi-cycle Implementation, Exceptions, Microprogramming: Simplifying Control Design, Introduction to Digital Design Using a Hardware Design Language.

Unit IV ***(10hours)***

Introduction to Pipelining, A pipelined Data path, Pipelined Control, Data Hazards and Forwarding, Data Hazards and Stalls, Branch Hazards, Exceptions, Advanced Pipelining.

Unit V ***(8hours)***

Introduction to Memory Hierarchy, The Basic of Caches, Measuring and Improving Cache Performance, Virtual Memory, Common Framework for Memory Hierarchies.

Unit VI (10hours)

Introduction to Storage, Networks and other Peripherals, Disk Storage and Dependability, Networks, Busses and other Connections between Processors, Memory and I/O Devices, Interfacing I/O Devices to the Processor, Memory and Operating System, I/O Performance Measures, Designing an I/O System.

Learning Resources

Text Books

1. DavidA.PattersonandJohnL.Hennessy*ComputerOrganizationandDesign* ', Morgan Kaufmann Publishers, 3rd Edition.

Reference Books

1. Ian McLoughlin '*Computer Architecture– An Embedded approach*',McGraw-Hill Education (Asia), 1stEdition.

Web resources

1. Prof AnshulKumar,NPTEL- IIT Delhi, '*Computer Architecture* ' .URL:http://nptel.ac.in/courses/106102062/

Course outcomes: At the end of the course, the student will be able to

1	Able to write recursive program in MIPS.
2	Able to construct cost effective computer system.
3	Able to differentiate different designs and organizations.
4	Able to handle design issues in the development of processor or other components that satisfies design requirements.

Assessment Method

Assessment Tool	Weeklytests (Insemester)	Monthly tests (In semester)	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

–

****

Course Code	Course Name	Course Category	L-T-P	Credits
20CSXX11 (ECE)	**Computer Networks**	**ESC**	**3: 0: 0**	**3**

Course Learning Objectives

1. To develop an understanding of modern network architectures from a design and performanceperspective.
2. To introduce the student to the major concepts involved in wide-area networks (WANs),local area networks (LANs) and Wireless LANs(WLANs).
3. To provide an opportunity to do networkprogramming
4. To provide a WLAN measurementideas.

Course Content

Unit -I **(8hours)**

Data communication Components: Representation of data and its flow Networks , Various Connection Topology, Protocols and Standards, OSI model, Transmission Media,

Unit-II ***(8 hours)***

LAN: Wired LAN, Wireless LANs, Connecting LAN and Virtual LAN, Techniques for Bandwidth utilization: Multiplexing - Frequency division, Time division and Wave division, Concepts on spreadspectrum.

Unit -III ***(8hours)***

Data Link Layer and Medium Access Sub Layer: Error Detection and Error Correction - Fundamentals, Block coding, Hamming Distance, CRC; Flow Control and Error control protocols - Stop and Wait, Go back–N ARQ, Selective Repeat ARQ, Sliding Window, Piggybacking, Random Access, Multiple access protocols -Pure ALOHA, Slotted ALOHA,CSMA/CD,CDMA/CA

Unit -IV ***(8hours)***

Network Layer: Switching, Logical addressing IPV4, IPV6; Address mapping ARP, RARP, BOOTP and DHCP–Delivery, Forwarding and Unicast Routingprotocols.

Unit -V ***(7hours)***

Transport Layer: Process to Process Communication, User Datagram Protocol (UDP), Transmission Control Protocol (TCP), SCTP Congestion Control; Quality of Service, QoS improving techniques: Leaky Bucket and Token Bucket algorithm.

Unit -VI *(6 hours)*

Application Layer: Domain Name Space (DNS), DDNS, TELNET, EMAIL, File Transfer Protocol (FTP), WWW, HTTP, SNMP, Bluetooth, Firewalls, Basic concepts of cryptography.

Learning resources Text book

1. Behrouz A.Forouzan *"Data Communications and Networking"*, 4e ,Tata McGraw Hill.
2. W. Stallings, "*Data and ComputerCommunication*", 8e, Pearson
3. Andrew S. Tanenbaum "*Computer Networks"*, 4e, PearsonEducation.

Reference Books

2. S.Kshev *"An Engineering Approach to Computer Networks "*, 2ndedition
3. W.A.Shay, Thomson *"Understanding Communications and Networks*",3rd edition,

Web resources

1. Prof AjitPal,NPTEL- IIT Kharagpur, 'Data Communications'.
   URL: https://nptel.ac.in/courses/106105082/
2. Prof Sujoy Gosh,NPTEL- IIT Kharagpur, 'Computer Networks'.
   URL:https://nptel.ac.in/courses/106105081/
3. https://www.tutorialspoint.com/computer_fundamentals/computer_networking

Course outcomes: At the end of the course, the student will be able to

CO 1	Explain the functions of the different layer of the OSI Protocol.
CO 2	Draw the functional block diagram of wide-area networks (WANs), local area networks (LANs) and Wireless LANs (WLANs) describe the function of each block.
CO 3	For a given requirement (small scale) of wide-area networks (WANs), local area networks (LANs) and Wireless LANs (WLANs) design it based on the market available component
CO 4	For a given problem related TCP/IP protocol Developed the network programming.

CO 5	Configure DNS DDNS, TELNET, EMAIL, File Transfer Protocol (FTP), WWW, HTTP, SNMP, Bluetooth, Firewalls using open sourceavailable software and tools.

Assessment method

Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

****

Course code	Course Name	Course Category	L-T-P	Credits
20CSXX07 (CHE)	**Object Oriented Programming through JAVA**	**ESC**	**3-0-0**	**3**

Course Learning Objectives:

1. Gain knowledge about basic Java language syntax and semantics to write Java programs and use concepts such as variables, conditional and iterative execution methods etc.,
2. Understanding the fundamentals of object-oriented programming in Java, including defining classes, objects, invoking methods etc. and exception handling mechanisms.
3. Understand the principles of inheritance, packages and interfaces.
4. Understand the principles of Multithreading and Event handling mechanisms.

Syllabus:

UNIT-1

Java Evolution, And Overview of Java Language: Java History –Features of java, how java differ from C and C++, Java and World Wide Web, Web Browser. Java Environment: Java Development kit (JDK), Application Programming Interface (API).

Java Programming Structure, Java Tokens, Constants, Variables, Expressions, Decision Making Statements and Looping, Java Statements, Overview of arrays and strings, Machine Neutral, Java Virtual Machine (JVM), Command Line Arguments

Arrays And Strings: Arrays, One-Dimensional arrays, creating an array, declaration of arrays, initialization of arrays, Two-Dimensional arrays, String arrays, String methods, String Buffer class, Vectors, Wrapper classes.

UNIT-2

Classes, Objects and Methods:

Introduction, defining a class, creating objects, accessing class members, constructors, methods overloading, static members.

Inheritance: Defining a sub class, sub class constructor, multilevel variables, Final classes, and Finalize methods, Abstract methods and classes, visibility control.

Managing Errors And Exceptions: Introduction, Types of errors-Compile time and Run time errors, Exceptions, Types of Exceptions, Syntax of Exception handling code, Multiple catch statements, Using finally statement, Throwing our own exceptions.

UNIT-3

Interfaces and Multithreaded Programming: Introduction, Defining interfaces, extending interfaces, implementing interfaces. Introduction to threads, Creating Threads, Extending the Thread Class, Implementing the runnable interface, life cycle of a thread, priority of a thread, synchronization, Dead Lock.

UNIT-4

Applet Programming

Introduction, how applet differ from applications, building applet code, applet life cycle, About HTML, designing a web page, passing parameters to applets, Getting input from the User.

UNIT-5

Graphics Programming

Introduction, the abstract window toolkit (AWT), frames, event-driven programming, layout managers, panels, canvasses, drawing geometric figures.

UNIT-6

Creating User Interface:

Introduction, Describe various user interface Components: button, label, text field, text area, choice, list, check box check box group.

Learning Resources

Text books:

1. Herbert Schildt, "The Complete Reference Java", TMH Publishing Company Ltd, 9th Edition.
2.Cay Horstmann, "Big Java", John Wiley and Sons, 2nd Edition

Reference Books:

1. Allen B. Downey, "Think Java; How to Think Like a Computer Scientist", Paper Back, 1stEdition
2. David J. Eck, Hobart and William Smith Colleges, "Introduction to Programming Using Java" Published by Paper Back
3. H.M.Dietel and P.J.Dietel "Java How to Program", Sixth Edition, Pearson Education/PHI

Web resources:

1. http://www.nptelvideos.com/java/java_video_lectures_tutorials.php
2. https://www.tutorialspoint.com/java/
3. https://www.javatpoint.com/java-tutorial
4. http://mooc.fi/courses/2013/programming-part-1/material.html
5. http://math.hws.edu/javanotes

Course Outcomes: At the end of the course, the student will be able to

CO 1	Explain OOP Principles and Write Basic Java Programs.
CO 2	Defining Classes and Objects. Identify classes, objects, members of a class and relationships among them needed for a specific problem
CO 3	To be able to write Java Programs to demonstrate method overloading and Demonstrate the concepts of polymorphism and inheritance. Discuss method overriding V/s method overloading.
CO 4	Explain the benefits of JAVA's Exceptional handling mechanism compared to other Programming Language
CO 5	To be able to write Java Programs to demonstrate Packages and Threading concepts.
CO 6	Discuss and Demonstrate the AWT Concepts and develop the AWT Applications.

Course Nature		**Theory**		
Assessment Method				
Assessment Tool	Weekly tests	Monthly tests	End Semester Test	Total
Weightage (%)	10%	30%	60%	100%

Course code	**Course Name**	**Course Category**	**L-T-P**	**Credits**
20CSXX87 (CHE)	**Object Oriented Programming though JAVA Laboratory**	**ESC**	**0: 0: 3**	**1.5 credits**

Course Learning Objective:

1. To build software development skills using java programming for real-world applications.
2. To understand and apply the concepts of classes, packages, interfaces, arraylist, User defined Linked List, File Handling, exception handling and Multi-threading.
3. To develop applications using AWT programming and event handling.

List of Experiments:
Lab No 1: Basic Programs in JAVA
Lab No 2: Programming Assignments on Arrays and Strings
Lab No 3: Programming Assignments on Classes, Objects and Encapsulation
Lab No 4: Implementing the concepts of Inheritance and Array Objects
Lab No 5: Implementing the OOPS Concepts of Abstract, Interfaces and Polymorphism
Lab No. 6: Programming Assignments on File Handling
Lab No. 7: Programming Exercises on Exception Handling
Lab No 8: Working with List Operations
Lab No 9: Implementing the concepts of Multi-Threading
Lab No 10: Programming Exercises on Event Handling

Course Outcomes
At the end of the course, the student will be able to

CO 1	Understanding the control structures and conditional statements in Java
CO 2	Understanding the arrays and String handling in java

CO 3	Understanding the difference between class and object and providing security for objects
CO 4	Understanding the reusability of objects and working with multiple objects
CO 5	Understanding about hiding the data, getting multiple inheritance through interfaces
CO 6	Understanding the data processing from files
CO 7	Understanding about handling run time abnormal program executions
CO 8	Understanding about creating user defined linked list and dynamic objects
CO 9	Understanding the multi-threaded programming and inter thread communication
CO 10	Understanding about GUI creation

Assessment Method

Assessment Tool	Experiments	Report/Viva-Voce/ Quiz/MCQ/Lab project	Total
Weightage (%)	25%	15%	40%
End Semester Examination weightage (%)			60%

****