COURSE STRUCTURE AND SYLLABUS

For

COMPUTER SCIENCE AND ENGINEERING
(Applicable for batches admitted from 2016-2017)

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY: KAKINADA
KAKINADA - 533 003, Andhra Pradesh, India

I Year - I Semester

S. No.	Subjects	L	T	P	Credits
1-HS	English – I	4	--	--	3
2-BS	Mathematics - I	4	--	--	3
3-BS	Mathematics – II (Mathematical Methods)	4	--	--	3
4-BS	Applied Physics	4	--	--	3
5	Computer Programming	4	--	--	3
6-ES	Engineering Drawing	4	--	--	3
7-HS	English - Communication Skills Lab - 1	--	--	3	2
8-BS	Applied / Engineering Physics Lab	--	--	3	2
9-ES	Applied / Engineering Physics – Virtual Labs – Assignments	--	--	2	--
10	Computer Programming Lab	--	--	3	2
Total Credits					**24**

I Year - II SEMESTER

S. No.	Subjects	L	T	P	Credits
1-HS	English – II	4	--	--	3
2-BS	Mathematics - III	4	--	--	3
3-BS	Applied Chemistry	4	--	--	3
4	Object Oriented Programming through C++	4	--	--	3
5-HS	Environmental Studies	4	--	--	3
6-ES	Engineering Mechanics	4	--	--	3
7-BS	Applied / Engineering Chemistry Laboratory	--	--	3	2
8-HS	English - Communication Skills Lab – 2	--	--	3	2
9	Object Oriented Programming Lab	--	--	3	2
Total Credits					**24**

II Year - I Semester

S. No.	Subjects	L	T	P	Credits
1-HS	Statistics with R Programming	4	--	--	3
2	Mathematical Foundations of Computer Science	4	--	--	3
3	Digital Logic Design	4	--	--	3
4	Python Programming	4	--	--	3
5	Data Structures through C++	4	--	--	3
6	Computer Graphics	4	--	--	3
7	Data Structures through C++Lab	--	--	3	2
8	Python Programming Lab	--	--	3	2
Total Credits					22

II Year - II Semester

S. No.	Subjects	L	T	P	Credits
1	Software Engineering	4	--	--	3
2	Java Programming	4	--	--	3
3	Advanced Data Structures	4	--	--	3
4	Computer Organization	4	--	--	3
5	Formal Languages and Automata Theory	4	--	--	3
6	Principles of Programming Languages	4	--	--	3
7	Advanced Data Structures Lab	--	--	3	2
8	Java Programming Lab	--	--	3	2
Total Credits					22

III Year - I Semester

S. No.	Subjects	L	T	P	Credits
1	Compiler Design	4	--	--	3
2	Unix Programming	4	--	--	3
3	Object Oriented Analysis and Design using UML	4	--	--	3
4	Database Management Systems	4	--	--	3
5	Operating Systems	4	--	--	3
6	Unified Modeling Lab	--	--	3	2
7	Operating System & Linux Programming Lab	--	--	3	2
8	Database Management System Lab	--	--	3	2
MC	Professional Ethics & Human Values	--	3	--	--
	Total Credits				**21**

III Year - II Semester

S. No.	Subjects	L	T	P	Credits
1	Computer Networks	4	2	--	3
2	Data Warehousing and Mining	4	--	--	3
3	Design and Analysis of Algorithms	4	--	--	3
4	Software Testing Methodologies	4	--	--	3
5	**Open Elective:** i. Artificial Intelligence ii. Internet of Things iii Cyber Security iv.Digital Signal Processing v.Embbeded Systems vi. Robotics	4	--	--	3
6	Network Programming Lab	--	--	3	2
7	Software Testing Lab	--	--	3	2
8	Data Warehousing and Mining Lab	--	--	3	2
9	IPR & Patents	--	2	--	--
	Total Credits				**21**

IV Year - I Semester

S. No.	Subjects	L	T	P	Credits
1	Cryptography and Network Security	4	--	--	3
2	Software Architecture & Design Patterns	4	--	--	3
3	Web Technologies	4	--	--	3
4- HS	Managerial Economics and Financial Analysis	4	--	--	3
5	**Elective-I** i. Big Data Analytics ii. Information Retrieval Systems iii. Mobile Computing	4	--	--	3
6	**Elective-II** i. Cloud Computing ii. Software Project Management iii. Scripting Languages	4	--	--	3
7	Software Architecture& Design Patterns Lab	--	--	3	2
8	Web Technologies Lab	--	--	3	2
	Total Credits				**22**

IV Year - II Semester

S. No.	Subjects	L	T	P	Credits
1	Distributed Systems	4	--	--	3
2- HS	Management Science	4	--	--	3
3	Machine Learning	4	--	--	3
4	**Elective-III** i.Concurrent and Parallel Programming ii.Artificial Neural Networks iii. Operations Research	4	--	--	3
5	Seminar	--	3	--	2
6	Project	--	--	--	10
	Total Credits				**24**

Total Course Credits =   48+44 + 42 + 46 = 180

<u>**SYLLABUS**</u>

I Year - I Semester

L	T	P	C
4	0	0	3

ENGLISH - I

Introduction:

In view of the growing importance of English as a tool for global communication and the consequent emphasis on training the students to acquire communicative competence, the syllabus has been designed to develop linguistic and communicative competence of the students of Engineering.

As far as the detailed Textbooks are concerned, the focus should be on the skills of listening, speaking, reading and writing. The nondetailed Textbooks are meant for extensive reading for pleasure and profit.

Thus the stress in the syllabus in primarily on the development of communicative skills and fostering of ideas.

Objectives:

1. To imporve the language proficiency of the students in English with emphasis on LSRW skills.
2. To enable the students to study and comprehend the prescribed lessons and subjects more effectively relating to their theorotical and practical components.
3. To develop the communication skills of the students in both formal and informal situations.

LISTENING SKILLS:

Objectives:

1. To enable the students to appreciate the role of listening skill and improve their pronounciation.
2. To enable the students to comprehend the speech of people belonging to different backgrounds and regions.
3. To enable the students to listen for general content, to fill up information and for specific information.

SPEAKING SKILLS:

Objectives:

1. To make the students aware of the importance of speaking for their personal and professional communication.
2. To enable the students to express themselves fluently and accurately in social and professional success.
3. To help the students describe objects, situations and people.
4. To make the students participate in group activities like roleplays, discussions and debates.
5. To make the students particiapte in Just a Minute talks.

READING SKILLS:

Objectives:

1. To enable the students to comprehend a text through silent reading.
2. To enable the students to guess the meanings of words, messages and inferences of texts in given contexts.
3. To enable the students to skim and scan a text.
4. To enable the students to identify the topic sentence.
5. To enable the students to identify discourse features.
6. To enable the students to make intensive and extensive reading.

WRITING SKILLS:

Objectives:

1. To make the students understand that writing is an exact formal skills.
2. To enable the students to write sentences and paragraphs.
3. To make the students identify and use appropriate vocabulary.
4. To enable the students to narrate and describe.
5. To enable the students capable of note-making.
6. To enable the students to write coherently and cohesively.
7. To make the students to write formal and informal letters.
8. To enable the students to describe graphs using expressions of comparision.
9. To enable the students to write techincal reports.

Methodology:

1. The class are to be learner-centered where the learners are to read the texts to get a comprehensive idea of those texts on their own with the help of the peer group and the teacher.
2. Integrated skill development methodology has to be adopted with focus on individual language skills as per the tasks/exercise.
3. The tasks/exercises at the end of each unit should be completed by the learners only and the teacher interventionis perimitted as per the complexity of the task/exercise.
4. The teacher is expected to use supplementary material wherever necessary and also generate activities/tasks as per the requirement.

5. The teacher is perimitted to use lecture method when a completely new concept is introduced in the class.

Assessment Procedure: Theory

1. The formative and summative assessment procedures are to be adopted (mid exams and end semester examination).
2. Neither the formative nor summative assessment procedures should test the memory of the content of the texts given in the textbook. The themes and global comprehension of the units in the present day context with application of the langauge skills learnt in the unit are to be tested.
3. Only new unseen passages are to be given to test reading skills of the learners. Written skills are to be tested from sentence level to essay level. The communication formats—emails,letters and reports-- are to be tested along with appropriate langauge and expressions.
4. Examinations:
I mid exam + II mid exam (15% for descriptive tests+10% for online tests)= 25%

(80% for the best of two and 20% for the other)

Assignments= 5%

End semester exams=70%

5. Three take home assignments are to be given to the learners where they will have to read texts from the reference books list or other sources and write their gist in their own words.

The following text books are recommended for study in I B.Tech I Semester (Common for all branches)and I B.Pharma I Sem of JNTU Kakinada from the academic year 2016-17

(R-16 Regulations)

DETAILED TEXTBOOK:

ENGLISH FOR ENGINEERS AND TECHNOLOGISTS, Published by **Orient Blackswan Pvt Ltd**

NON-DETAILED TEXTBOOK:

PANORAMA: A COURSE ON READING, Published by **Oxford University Press India**

The course content along with the study material is divided into six units.

UNIT I:

1. 'Human Resources' from English for Engineers and Technologists.

OBJECTIVE:

To develop human resources to serve the society in different ways.

OUTCOME:

The lesson motivates the readers to develop their knowledge different fields and serve the society accordingly.

2. 'An Ideal Family' from Panorama: A Course on Reading

OBJECTIVE:

To develop extensive reading skill and comprehension for pleasure and profit.

OUTCOME:

Acquisition of writing skills

UNIT 2:

1. ' Transport: Problems and Solutions' from English for Engineers and Technologists.

OBJECTIVE:

To highlight road safety measures whatever be the mode of transport.

OUTCOME:

The lesson motivates the public to adopt road safety measures.

2. 'War' from 'Panorama : A Course on Reading'

OBJECTIVE:

To develop extensive reading skill and comprehension for pleasure and profit.

OUTCOME:

Acquisition of writing skills

UNIT 3:

1. 'Evaluating Technology' from English for Engineers and Technologists.

OBJECTIVE:

To highlight the advantages and disadvantages of technology.

OUTCOME:

The lesson creates an awareness in the readers that mass production is ultimately detrimental to biological survival.

2. 'The Verger' from 'Panorama : A Course on Reading'

OBJECTIVE:

To develop extensive reading skill and comprehension for pleasure and profit.

OUTCOME:

Acquisition of writing skills

UNIT 4:

1. 'Alternative Sources of Energy' from English for Engineers and Technologists.

OBJECTIVE:

To bring into focus different sources of energy as alternatives to the depleting sources.

OUTCOME:

The lesson helps to choose a source of energy suitable for rural India.

2. ' The Scarecrow' from Panorama : A Course on Reading

OBJECTIVE:

To develop extensive reading skill and comprehension for pleasure and profit.

OUTCOME:

Acquisition of writing skills

UNIT 5:

1. 'Our Living Environment' from English for Engineers and Technologists.

OBJECTIVE:

To highlight the fact that animals must be preserved beacuase animal life is precious.

OUTCOME:

The lesson creates an awareness in the reader as to the usefulness of animals for the human society.

2. 'A Village Host to Nation' from Panorama : A Course on Reading

OBJECTIVE:

To develop extensive reading skill and comprehension for pleasure and profit.

OUTCOME:

Acquisition of writing skills

UNIT 6:

1. ' Safety and Training' from English for Engineers and Technologists.

OBJECTIVE:

To highlight the possibility of accidents in laboratories, industries and other places and to follow safety measures.

OUTCOME:

The lesson helps in identifying safety measures against different varieties of accidents at home and in the workplace.

2. 'Martin Luther King and Africa' from Panorama : A Course on Reading

OBJECTIVE:

To develop extensive reading skill and comprehension for pleasure and profit.

OUTCOME:

Acquisition of writing skills

NOTE:

   All the exercises given in the prescribed lessons in both detailed and non-detailed textbooks relating to the theme and language skills must be covered.

OVERALL COURSE OUTCOME:

   1. Using English languages, both written and spoken, competently and correctly.
   2. Improving comprehension and fluency of speech.
   3. Gaining confidence in using English in verbal situations.

MODEL QUESTION PAPER FOR THEORY

PART- I

Six short answer questions on 6 unit themes

One question on eliciting student's response to any of the themes

PART-II

Each question should be from one unit and the last question can be a combination of two or more units.

Each question should have 3 sub questions: A,B & C

A will be from the main text: 5 marks

B from non-detailed text: 3 marks

C on grammar and Vocabulary: 6 marks

L	T	P	C
4	0	0	3

MATHEMATICS-I
(Common to ALL branches of First Year B.Tech.)

Course Objectives:
1. The course is designed to equip the students with the necessary mathematical skills and techniques that are essential for an engineering course.
2. The skills derived from the course will help the student from a necessary base to develop analytic and design concepts.

Course Outcomes: At the end of the Course, Student will be able to:
1. Solve linear differential equations of first, second and higher order.
2. Determine Laplace transform and inverse Laplace transform of various functions and use Laplace transforms to determine general solution to linear ODE.
3. Calculate total derivative, Jocobian and minima of functions of two variables.

UNIT I: Differential equations of first order and first degree:
Linear-Bernoulli-Exact-Reducible to exact.
Applications: Newton's Law of cooling-Law of natural growth and decay-Orthogonal trajectories- Electrical circuits- Chemical reactions.

UNIT II: Linear differential equations of higher order:
Non-homogeneous equations of higher order with constant coefficients with RHS term of the type e^{ax}, sin ax, cos ax, polynomials in x, e^{ax} V(x), xV(x)- Method of Variation of parameters.
Applications: LCR circuit, Simple Harmonic motion.

UNIT III: Laplace transforms:
Laplace transforms of standard functions-Shifting theorems - Transforms of derivatives and integrals – Unit step function –Dirac's delta function- Inverse Laplace transforms– Convolution theorem (with out proof).
Applications: Solving ordinary differential equations (initial value problems) using Laplace transforms.

UNIT IV: Partial differentiation:
Introduction- Homogeneous function-Euler's theorem-Total derivative-Chain rule-Generalized Mean value theorem for single variable (without proof)-Taylor's and Mc Laurent's series expansion of functions of two variables– Functional dependence- Jacobian.
Applications: Maxima and Minima of functions of two variables without constraints and Lagrange's method (with constraints).

UNIT V: First order Partial differential equations:
Formation of partial differential equations by elimination of arbitrary constants and arbitrary functions –solutions of first order linear (Lagrange) equation and nonlinear (standard types) equations.

UNIT VI: Higher order Partial differential equations:
Solutions of Linear Partial differential equations with constant coefficients. RHS term of the type $e^{ax+by}, \sin(ax+by), \cos(ax+by), x^m y^n$. Classification of second order partial differential equations.

Text Books:

1. **B.S.Grewal,** Higher Engineering Mathematics, 43rd Edition, Khanna Publishers.
2. **N.P.Bali**, Engineering Mathematics, Lakshmi Publications.

Reference Books:

1. **Erwin Kreyszig,** Advanced Engineering Mathematics, 10th Edition, Wiley-India
2. **Micheael Greenberg,** Advanced Engineering Mathematics, 9th edition, Pearson edn
3. **Dean G. Duffy,** Advanced engineering mathematics with MATLAB, CRC Press
4. **Peter O'neil,** Advanced Engineering Mathematics, Cengage Learning.
5. **Srimanta Pal, Subodh C.Bhunia**, Engineering Mathematics, Oxford University Press.
6. **Dass H.K., Rajnish Verma. Er.,** Higher Engineering Mathematics, S. Chand Co. Pvt. Ltd, Delhi.

L	T	P	C
4	0	0	3

I Year - I Semester

MATHEMATICS-II (Mathematical Methods)
(Common to ALL branches of First Year B.Tech.)

Course Objectives:
1. The course is designed to equip the students with the necessary mathematical skills and techniques that are essential for an engineering course.
2. The skills derived from the course will help the student from a necessary base to develop analytic and design concepts.
3. Understand the most basic numerical methods to solve simultaneous linear equations.

Course Outcomes: At the end of the Course, Student will be able to:
1. Calculate a root of algebraic and transcendental equations. Explain relation between the finite difference operators.
2. Compute interpolating polynomial for the given data.
3. Solve ordinary differential equations numerically using Euler's and RK method.
4. Find Fourier series and Fourier transforms for certain functions.
5. Identify/classify and solve the different types of partial differential equations.

UNIT I: Solution of Algebraic and Transcendental Equations:
Introduction- Bisection method – Method of false position – Iteration method – Newton-Raphson method (One variable and simultaneous Equations).

UNIT II: Interpolation:
Introduction- Errors in polynomial interpolation – Finite differences- Forward differences-Backward differences –Central differences – Symbolic relations and separation of symbols - Differences of a polynomial-Newton's formulae for interpolation – Interpolation with unequal intervals - Lagrange's interpolation formula.

UNIT III: Numerical Integration and solution of Ordinary Differential equations:
Trapezoidal rule- Simpson's $1/3^{rd}$ and $3/8^{th}$ rule-Solution of ordinary differential equations by Taylor's series-Picard's method of successive approximations-Euler's method - Runge-Kutta method (second and fourth order).

UNIT IV: Fourier Series:
Introduction- Periodic functions – Fourier series of -periodic function - Dirichlet's conditions – Even and odd functions –Change of interval– Half-range sine and cosine series.

UNIT V: Applications of PDE:
Method of separation of Variables- Solution of One dimensional Wave, Heat and two-dimensional Laplace equation.

UNIT VI: Fourier Transforms:

Fourier integral theorem (without proof) – Fourier sine and cosine integrals - sine and cosine transforms – properties – inverse transforms – Finite Fourier transforms.

Text Books:

1. **B.S.Grewal,** Higher Engineering Mathematics, 43rd Edition, Khanna Publishers.
2. **N.P.Bali**, Engineering Mathematics, Lakshmi Publications.

Reference Books:

1. **Dean G. Duffy,** Advanced engineering mathematics with MATLAB, CRC Press
2. **V.Ravindranath and P.Vijayalakshmi,** Mathematical Methods, Himalaya Publishing House.
3. **Erwin Kreyszig,** Advanced Engineering Mathematics, 10th Edition, Wiley-India
4. **David Kincaid, Ward Cheney**, Numerical Analysis-Mathematics of Scientific Computing, 3rd Edition, Universities Press.
5. **Srimanta Pal, Subodh C.Bhunia**, Engineering Mathematics, Oxford University Press.
6. **Dass H.K., Rajnish Verma. Er.,** Higher Engineering Mathematics, S. Chand Co. Pvt. Ltd, Delhi.

L	T	P	C
4	0	0	3

I Year - I Semester

APPLIED PHYSICS
(CSE, ECE, EEE, IT, EIE, E.Com.E)

OBJECTIVES: _Physics curriculum which is re-oriented to the needs of Circuital branches of graduate engineering courses offered by JNTUniv.Kkd. that serves as a transit to understand the branch specific advanced topics. The courses are designed to:_

- _Impart Knowledge of Physical Optics phenomena like Interference, Diffraction and Polarization involving required to design instruments with higher resolution._
- _Teach Concepts of coherent sources, its realization and utility optical instrumentation._
- _Study the concepts regarding the bulk response of materials to the EM fields and their analytically study in the back-drop of basic quantum mechanics._
- _Understand the physics of Semiconductors and their working mechanism for their utility in sensors._

UNIT-I
INTERFERENCE: Principle of Superposition – Coherent Sources – Interference in thin films (reflection geometry) – Newton's rings – construction and basic principle of Interferometers.

UNIT-II
DIFFRACTION: Fraunhofer diffraction at single slit - Cases of double slit, N-slits & Circular Aperture (Qualitative treatment only)-Grating equation - Resolving power of a grating, Telescope and Microscopes.

UNIT-III
POLARIZATION: Types of Polarization – Methods of production - Nicol Prism - Quarter wave plate and Half Wave plate – Working principle of Polarimeter (Sacharimeter).
LASERS: Characteristics– Stimulated emission – Einstein's Transition Probabilities-Pumping schemes - Ruby laser – Helium Neon laser.

UNIT-IV
ELECTROMAGNETIC FIELDS: Scalar and Vector Fields – Electric Potential-Gradient, Divergence of fields – Gauss and Stokes theorems-Propagation of EM waves through dielectric medium.

UNIT-V
QUANTUM MECHANICS: Introduction - Matter waves – Schröedinger Time Independent and Time Dependent wave equations – Particle in a box.
FREE ELECTRON THEORY: Defects of Classical free electron theory –Quantum Free electron theory - concept of Fermi Energy.

UNIT-VI

BAND THEORY OF SOLIDS: Bloch's theorem (qualitative) – Kronig – Penney model – energy bands in crystalline solids – classification of crystalline solids– effective mass of electron & concept of hole.

SEMICONDUCTOR PHYSICS: Conduction – Density of carriers in Intrinsic and Extrinsic semiconductors – Drift & Diffusion – relevance of Einstein's equation- Hall effect in semiconductors

Outcome: Construction and working details of instruments, ie., Interferometer, Diffractometer and Polarimeter are learnt. Study EM-fields and semiconductors under the concepts of Quantum mechanics paves way for their optimal utility.

Text Books:

1. A Text book of Engineering Physics – by Dr. M.N.Avadhanulu and Dr.P.G.Kshira sagar, S.Chand & Company Ltd., (2014)
2. 'Solid State Physics' by A.J.Dekker, Mc Millan Publishers (2011)
3. Engineering Physics by D.K.Bhattacharya and Poonam Tandon, Oxford press (2015)

Reference Books:

1. Applied Physics by P.K.Palanisamy, Scitech publications (2014)
2. Lasers and Non-Linear optics by B.B.Laud, New Age International Publishers (2008).
3. Engineering Physics by M. Arumugam, Anuradha Publication (2014)

L	T	P	C
4	0	0	3

COMPUTER PROGRAMMING

Learning objectives:
Formulating algorithmic solutions to problems and implementing algorithms in C.
- Notion of Operation of a CPU, Notion of an algorithm and computational procedure, editing and executing programs in Linux.
- Understanding branching, iteration and data representation using arrays.
- Modular programming and recursive solution formulation.
- Understanding pointers and dynamic memory allocation.
- Understanding miscellaneous aspects of C.
- Comprehension of file operations.

UNIT-I:
History and Hardware - Computer Hardware, Bits and Bytes, Components, Programming Languages - Machine Language, Assembly Language, Low- and High-Level Languages, Procedural and Object-Oriented Languages, Application and System Software, The Development of C Algorithms The Software Development Process.

UNIT-II:
Introduction to C Programming- Identifiers, The main () Function, The printf () Function
Programming Style - Indentation, Comments, Data Types, Arithmetic Operations, Expression Types, Variables and Declarations, Negation, Operator Precedence and Associativity, Declaration Statements, Initialization.
Assignment - Implicit Type Conversions, Explicit Type Conversions (Casts), Assignment Variations, Mathematical Library Functions, Interactive Input, Formatted Output, Format Modifiers.

UNIT -III:
Control Flow-Relational Expressions - Logical Operators:
Selection: if-else Statement, nested if, examples, Multi-way selection: switch, else-if, examples.
Repetition: Basic Loop Structures, Pretest and Posttest Loops, Counter-Controlled and Condition-Controlled Loops, The while Statement, The for Statement, Nested Loops, The do-while Statement.

UNIT-IV
Modular Programming: Function and Parameter Declarations, Returning a Value, Functions with Empty Parameter Lists, Variable Scope, Variable Storage Class, Local Variable Storage Classes, Global Variable Storage Classes, Pass by Reference, Passing Addresses to a Function, Storing Addresses, Using Addresses, Declaring and Using Pointers, Passing Addresses to a Function.
Case Study: Swapping Values, Recursion - Mathematical Recursion, Recursion versus Iteration.

UNIT-V:
Arrays & Strings
Arrays: One-DimensionalArrays, Input and Output of Array Values, Array Initialization, Arrays as Function Arguments, Two-Dimensional Arrays, LargerDimensionalArrays- Matrices
Strings: String Fundamentals, String Input and Output, String Processing, Library Functions

UNIT-VI:
Pointers, Structures, Files
Pointers: Concept of a Pointer, Initialisation of pointer variables, pointers as function arguments, passing by address, Dangling memory, address arithmetic, character pointers and functions, pointers to pointers, Dynamic memory management functions, command line arguments.
Structures: Derived types,Structuresdeclaration, Initialization of structures, accessing structures, nested structures, arrays of structures, structures and functions, pointers to structures, self referential structures, unions, typedef, bit-fields.
Data Files: Declaring, Opening, and Closing File Streams, Reading from and Writing to Text Files, Random File Access

Outcomes:
- Understand the basic terminology used in computer programming
- Write, compile and debug programs in C language.
- Use different data types in a computer program.
- Design programs involving decision structures, loops and functions.
- Explain the difference between call by value and call by reference
- Understand the dynamics of memory by the use of pointers
- Use different data structures and create/update basic data files.

Text Books:
1. ANSI C Programming, Gary J. Bronson, Cengage Learning.
2. Programming in C, Bl Juneja Anita Seth, Cengage Learning.
3. The C programming Language, Dennis Richie and Brian Kernighan, Pearson Education.

Reference Books:
1. C Programming-A Problem Solving Approach, Forouzan, Gilberg, Cengage.
2. Programming with C, Bichkar, Universities Press.
3. Programming in C, ReemaThareja, OXFORD.
4. C by Example, Noel Kalicharan, Cambridge.

I Year - I Semester

L	T	P	C
4	0	0	3

ENGINEERING DRAWING

Objective: Engineering drawing being the principle method of communication for engineers, the objective to introduce the students, the techniques of constructing the various types of polygons, curves and scales. The objective is also to visualize and represent the 3D objects in 2D planes with proper dimensioning, scaling etc.

- To introduce the use and the application of drawing instruments and to make the students construct the polygons, curves and various types of scales. The student will be able to understand the need to enlarge or reduce the size of objects in representing them.
- To introduce orthographic projections and to project the points and lines parallel to one plane and inclined to other.
- To make the students draw the projections of the lines inclined to both the planes.
- To make the students draw the projections of the plane inclined to both the planes.
- To make the students draw the projections of the various types of solids in different positions inclined to one of the planes.
- To represent the object in 3D view through isometric views. The student will be able to represent and convert the isometric view to orthographic view and vice versa.

UNIT I Polygons, Construction of regular polygons using given length of a side; Ellipse, arcs of circles and Oblong methods; Scales – Vernier and Diagonal scales.

UNIT II Introduction to orthographic projections; projections of points; projections of straight lines parallel to both the planes; projections of straight lines – parallel to one plane and inclined to the other plane.

UNIT III Projections of straight lines inclined to both the planes, determination of true lengths, angle of inclinations and traces.

UNIT IV Projections of planes: regular planes perpendicular/parallel to one plane and inclined to the other reference plane; inclined to both the reference planes.

UNIT V Projections of Solids – Prisms, Pyramids, Cones and Cylinders with the axis inclined to one of the planes.

UNIT VI Conversion of isometric views to orthographic views; Conversion of orthographic views to isometric views.

Text Books:

1. Engineering Drawing, N. D. Butt, Chariot Publications

2. Engineering Drawing, K. L. Narayana & P. Kannaiah, Scitech Publishers.

3. Engineering Graphics, P.I. Varghese, McGraw Hill Publishers

Reference Books:

1. Engineering Graphics for Degree, K. C. John, PHI Publishers

2. Engineering Drawing, Agarwal & Agarwal, Tata McGraw Hill Publishers

3. Engineering Drawing + AutoCad – K Venugopal, V. Prabhu Raja, New Age

L	T	P	C
0	0	3	2

ENGLISH - COMMUNICATION SKILLS LAB - I

PRESCRIBED LAB MANUAL FOR SEMESTER I:

'INTERACT: English Lab Manual for Undergraduate Students' Published by **Orient Blackswan Pvt Ltd.**

OBJECTIVES:

To enable the students to learn through practice the communication skills of listening, speaking, reading and writing.

OUTCOME:

A study of the communicative items in the laboratory will help the students become successful in the competitive world.

The course content along with the study material is divided into six units.

UNIT 1:

1. WHY study Spoken English?
2. Making Inqueries on the phone, thanking and responding to Thanks
   Practice work.

UNIT 2:

1. Responding to Requests and asking for Directions
   Practice work.

UNIT 3:

1. Asking for Clarifications, Inviting, Expressing Sympathy, Congratulating
2. Apologising, Advising, Suggesting, Agreeing and Disagreeing
   Practice work.

UNIT 4:

1. Letters and Sounds
   Practice work.

UNIT 5:

1. The Sounds of English
   Practice work.

UNIT 6:

1. Pronunciation
2. Stress and Intonation
   Practice work.

Assessment Procedure: Laboratory

1. Every lab session (150 minutes) should be handled by not less than two teachers (three would be ideal) where each faculty has to conduct a speaking activity for 20/30 students.
2. The teachers are to assess each learner in the class for not less than 10 speaking activities, each one to be assessed for 10 marks or 10%. The average of 10 day-to-day activity assessments is to be calculated for 10 marks for internal assessment.

The rubric given below has to be filled in for all the students for all activities.

The rubric to assess the learners:

	Body language		Fluency & Audibility	Clarity in Speech	Neutraliz ation of accent	Appropriate Language		Total 10 marks	Remarks
	Gestures & Postures	Eye Conta ct				Gramm ar	Vocabu lary & expressi ons		

- **Lab Assessment: Internal (25 marks)**
  1. Day-to-Day activities: 10 marks

  2. Completing the exercises in the lab manual: 5 marks

  3. Internal test (5 marks written and 5 marks oral)

- **Lab Assessment: External (50 marks)**
  1. Written test: 20 marks (writing a dialogue, note-taking and answering questions on listening to an audio recording.

  2. Oral: Reading aloud a text or a dialogue- 10 marks

  3. Viva-Voce by the external examiner: 20 marks

Reference Books:

1. Strengthen your communication skills by Dr M Hari Prasad, Dr Salivendra Raju and Dr G Suvarna Lakshmi, Maruti Publications.
2. English for Professionals by Prof Eliah, B.S Publications, Hyderabad.
3. Unlock, Listening and speaking skills 2, Cambridge University Press
4. Spring Board to Success, Orient BlackSwan
5. A Practical Course in effective english speaking skills, PHI
6. Word power made handy, Dr shalini verma, Schand Company
7. Let us hear them speak, Jayashree Mohanraj, Sage texts
8. Professional Communication, Aruna Koneru, Mc Grawhill Education
9. Cornerstone, Developing soft skills, Pearson Education

	L	T	P	C
I Year - I Semester	0	0	3	2

APPLIED/ENGINEERING PHYSICS LAB

(Any 10 of the following listed experiments)

Objective: *Training field oriented Engineering graduates to handle instruments and their design methods to improve the accuracy of measurements.*

LIST OF EXPERIMENTS:

1. Determination of wavelength of a source-Diffraction Grating-Normal incidence.

2. Newton's rings – Radius of Curvature of Plano - Convex Lens.

3. Determination of thickness of a spacer using wedge film and parallel interference fringes.

4. Determination of Rigidity modulus of a material- Torsional Pendulum.

5. Determination of Acceleration due to Gravity and Radius of Gyration- Compound Pendulum.

6. Melde's experiment – Transverse and Longitudinal modes.

7. Verification of laws of vibrations in stretched strings – Sonometer.

8. Determination of velocity of sound – Volume Resonator.

9. L- C- R Series Resonance Circuit.

10. Study of I/V Characteristics of Semiconductor diode.

11. I/V characteristics of Zener diode.

12. Characteristics of Thermistor – Temperature Coefficients.

13. Magnetic field along the axis of a current carrying coil – Stewart and Gee's apparatus.

14. Energy Band gap of a Semiconductor p - n junction.

15. Hall Effect in semiconductors.

16. Time constant of CR circuit.

17. Determination of wavelength of laser source using diffraction grating.

18. Determination of Young's modulus by method of single cantilever oscillations.

19. Determination of lattice constant – lattice dimensions kit.

20. Determination of Planck's constant using photocell.

21. Determination of surface tension of liquid by capillary rise method.

Outcome: *Physics lab curriculum gives fundamental understanding of design of an instrument with targeted accuracy for physical measurements*

	L	T	P	C
	0	0	2	0

I Year - I Semester

APPLIED/ENGINEERING PHYSICS - VIRTUAL LABS – ASSIGNMENTS
(Constitutes 5% marks of 30marks of Internal-component)

Objective: *Training Engineering students to prepare a technical document and improving their writing skills.*

LIST OF EXPERIMENTS

1. Hall Effect
2. Crystal Structure
3. Hysteresis
4. Brewster's angle
5. Magnetic Levitation / SQUID
6. Numerical Aperture of Optical fiber
7. Photoelectric Effect
8. Simple Harmonic Motion
9. Damped Harmonic Motion
10. LASER – Beam Divergence and Spot size
11. B-H curve
12. Michelson's interferometer
13. Black body radiation

URL: www.vlab.co.in

Outcome: *Physics Virtual laboratory curriculum in the form of assignment ensures an engineering graduate to prepare a /technical/mini-project/ experimental report with scientific temper.*

L	T	P	C
0	0	3	2

I Year - I Semester

COMPUTER PROGRAMMING LAB

OBJECTIVES:
• Understand the basic concept of C Programming, and its different modules that includes conditional and looping expressions, Arrays, Strings, Functions, Pointers, Structures and File programming.

• Acquire knowledge about the basic concept of writing a program.

• Role of constants, variables, identifiers, operators, type conversion and other building blocks of C Language.

• Use of conditional expressions and looping statements to solve problems associated with conditions and repetitions.

• Role of Functions involving the idea of modularity.

Programming

Exercise - 1 Basics
a) What is an OS Command, Familiarization of Editors - vi, Emacs
b) Using commands like mkdir, ls, cp, mv, cat, pwd, and man
c) C Program to Perform Adding, Subtraction, Multiplication and Division of two numbers From Command line

Exercise - 2 Basic Math
a) Write a C Program to Simulate 3 Laws at Motion
b) Write a C Program to convert Celsius to Fahrenheit and vice versa

Exercise - 3 Control Flow - I
a)Write a C Program to Find Whether the Given Year is a Leap Year or not.
b)Write a C Program to Add Digits & Multiplication of a number

Exercise – 4 Control Flow - II
a)Write a C Program to Find Whether the Given Number is
        i) Prime Number
        ii) Armstrong Number
b) Write a C program to print Floyd Triangle
c) Write a C Program to print Pascal Triangle

Exercise – 5 Functions
a) Write a C Program demonstrating of parameter passing in Functions and returning values.
b) Write a C Program illustrating Fibonacci, Factorial with Recursion without Recursion

Exercise – 6 Control Flow - III
a) Write a C Program to make a simple Calculator to Add, Subtract, Multiply or Divide Using switch…case

b) Write a C Program to convert decimal to binary and hex (using switch call function the function)

Exercise – 7 Functions - Continued
Write a C Program to compute the values ofsin x and cos x and e^x values using Series expansion. (use factorial function)

Exercise – 8 Arrays
Demonstration of arrays
a) Search-Linear.
b) Sorting-Bubble, Selection.
c) Operations on Matrix.

Exercises - 9 Structures
a)Write a C Program to Store Information of a Movie Using Structure
b)Write a C Program to Store Information Using Structures with Dynamically Memory Allocation
c) Write a C Program to Add Two Complex Numbers by Passing Structure to a Function

Exercise - 10 Arrays and Pointers
a)Write a C Program to Access Elements of an Array Using Pointer
b) Write a C Program to find the sum of numbers with arrays and pointers.

Exercise – 11 Dynamic Memory Allocations
 a)   Write a C program to find sum of n elements entered by user. To perform this program, allocate memory dynamically using malloc () function.

b) Write a C program to find sum of n elements entered by user. To perform this program, allocate memory dynamically using calloc () function. Understand the difference between the above two programs

Exercise – 12 Strings
a) Implementation of string manipulation operations **with** library function.
        i) copy
        ii) concatenate
        iii) length
        iv) compare
b) Implementation of string manipulation operations **without** library function.
        i) copy
        ii) concatenate
        iii) length
        iv) compare

Exercise -13 Files
a)Write a C programming code to open a file and to print it contents on screen.
b)Write a C program to copy files

Exercise - 14 Files Continued
a) Write a C program merges two files and stores their contents in another file.
b) Write a C program to delete a file.

Exercise - 15

a) System Assembling, Disassembling and identification of Parts / Peripherals. b) Operating System Installation-Install Operating Systems like Windows, Linux along with necessary Device Drivers.

Exercise - 16

    a) MS-Office / Open Office

      i) Word - Formatting, Page Borders, Reviewing, Equations, symbols.   ii) Spread Sheet - organize data, usage of formula, graphs, charts.

      iii) Powerpoint - features of power point, guidelines for preparing an effective presentation.

    b) Network Configuration & Software Installation-Configuring TCP/IP, Proxy, and firewall settings. Installing application software, system software & tools.

OUTCOMES:

• Apply and practice logical ability to solve the problems.

• Understand C programming development environment, compiling, debugging, and linking and executing a program using the development environment

• Analyzing the complexity of problems, Modularize the problems into small modules and then convert them into programs

• Understand and apply the in-built functions and customized functions for solving the problems.

• Understand and apply the pointers, memory allocation techniques and use of files for dealing with variety of problems.

• Document and present the algorithms, flowcharts and programs in form of user-manuals

•Identification of various computer components, Installation of software

Note:

a) All the Programs must be executed in the Linux Environment. (Mandatory)
b) The Lab record must be a print of the LATEX (.tex) Format.

	L	T	P	C
	4	0	0	3

I Year - II Semester

ENGLISH -II

Introduction:

In view of the growing importance of English as a tool for global communication and the consequent emphasis on training the students to acquire communicative competence, the syllabus has been designed to develop linguistic and communicative competence of the students of Engineering.

As far as the detailed Textbooks are concerned, the focus should be on the skills of listening, speaking, reading and writing. The nondetailed Textbooks are meant for extensive reading for pleasure and profit.

Thus the stress in the syllabus in primarily on the development of communicative skills and fostering of ideas.

Objectives:

1. To imporve the language proficiency of the students in English with emphasis on LSRW skills.
2. To enable the students to study and comprehend the prescribed lessons and subjects more effectively relating to their theorotical and practical components.
3. To develop the communication skills of the students in both formal and informal situations.

LISTENING SKILLS:

Objectives:

1. To enable the students to appreciate the role of listening skill and improve their pronounciation.
2. To enable the students to comprehend the speech of people belonging to different backgrounds and regions.
3. To enable the students to listen for general content, to fill up information and for specific information.

SPEAKING SKILLS:

Objectives:

1. To make the students aware of the importance of speaking for their personal and professional communication.
2. To enable the students to express themselves fluently and accurately in social and professional success.
3. To help the students describe objects, situations and people.
4. To make the students participate in group activities like roleplays, discussions and debates.
5. To make the students particiapte in Just a Minute talks.

READING SKILLS:

Objectives:

1. To enable the students to comprehend a text through silent reading.
2. To enable the students to guess the meanings of words, messages and inferences of texts in given contexts.
3. To enable the students to skim and scan a text.
4. To enable the students to identify the topic sentence.
5. To enable the students to identify discourse features.
6. To enable the students to make intensive and extensive reading.

WRITING SKILLS:

Objectives:

1. To make the students understand that writing is an exact formal skills.
2. To enable the students to write sentences and paragraphs.
3. To make the students identify and use appropriate vocabulary.
4. To enable the students to narrate and describe.
5. To enable the students capable of note-making.
6. To enable the students to write coherently and cohesively.
7. To make the students to write formal and informal letters.
8. To enable the students to describe graphs using expressions of comparision.
9. To enable the students to write techincal reports.

Methodology:

1. The class are to be learner-centered where the learners are to read the texts to get a comprehensive idea of those texts on their own with the help of the peer group and the teacher.
2. Integrated skill development methodology has to be adopted with focus on individual language skills as per the tasks/exercise.
3. The tasks/exercises at the end of each unit should be completed by the learners only and the teacher interventionis perimitted as per the complexity of the task/exercise.
4. The teacher is expected to use supplementary material wherever necessary and also generate activities/tasks as per the requirement.

5. The teacher is perimitted to use lecture method when a completely new concept is introduced in the class.

Assessment Procedure: Theory

1. The formative and summative assessment procedures are to be adopted (mid exams and end semester examination).
2. Neither the formative nor summative assessment procedures should test the memory of the content of the texts given in the textbook. The themes and global comprehension of the units in the present day context with application of the langauge skills learnt in the unit are to be tested.
3. Only new unseen passages are to be given to test reading skills of the learners. Written skills are to be tested from sentence level to essay level. The communication formats— emails,letters and reports-- are to be tested along with appropriate langauge and expressions.
4. Examinations:

   I mid exam + II mid exam (15% for descriptive tests+10% for online tests)= 25%

   (80% for the best of two and 20% for the other)

   Assignments= 5%

   End semester exams=70%

5. Three take home assignments are to be given to the learners where they will have to read texts from the reference books list or other sources and write their gist in their own words.

The following text books are recommended for study in I B.Tech II Semester (Common for all branches)and I B.Pharma II Sem of JNTU Kakinada from the academic year 2016-17 **(R-16 Regulations)**

DETAILED TEXTBOOK: ENGLISH ENCOUNTERS Published by **Maruthi Publishers**.

DETAILED NON-DETAIL:THE GREAT INDIAN SCIENTISTS Published by **Cenguage learning**

The course content along with the study material is divided into six units.

UNIT 1:

1. ' The Greatest Resource- Education' from English Encounters

OBJECTIVE:

Schumacher describes the education system by saying that it was mere training, something more than mere knowledge of facts.

OUTCOME:

The lesson underscores that the ultimate aim of Education is to enhance wisdom.

2. ' A P J Abdul Kalam' from The Great Indian Scientists.

OBJECTIVE:

The lesson highlights Abdul Kalam's contributions to Indian science and the awards he received.

OUTCOME:

Abdul Kalam's simple life and service to the nation inspires the readers to follow in his footsteps.

UNIT 2:

1. ' A Dilemma' from English Encounters

OBJECTIVE: The lesson centres on the pros and cons of the development of science and technology.

OUTCOME: The lesson enables the students to promote peaceful co-existence and universal harmony among people and society.

2. 'C V Raman' from The Great Indian Scientists.

OBJECTIVE:

The lesson highlights the dedicated research work of C V Raman and his achievements in Physics.

OUTCOME:

The Achievements of C V Raman are inspiring and exemplary to the readers and all scientists.

UNIT 3:

1. 'Cultural Shock' Adjustments to new Cultural Environments from English Encounters.

OBJECTIVE:

The lesson depicts of the symptoms of Cultural Shock and the aftermath consequences.

OUTCOME: The lesson imparts the students to manage different cultural shocks due to globalization.

2. 'Homi Jehangir Bhabha' from The Great Indian Scientists.

OBJECTIVE:

The lesson highlights Homi Jehangir Bhabha's contributions to Indian nuclear programme as architect.

OUTCOME:

The seminal contributions of Homi Jehangir Bhabha to Indian nuclear programme provide an aspiration to the readers to serve the nation and sterngthen it.

UNIT 4:

1. 'The Lottery' from English Encounters.

OBJECTIVE:

The lesson highlights insightful commentary on cultural traditions.

OUTCOME:

The theme projects society's need to re examine its traditions when they are outdated.

2. 'Jagadish Chandra Bose' from The Great Indian Scientists.

OBJECTIVE:

The lesson gives an account of the unique discoveries and inventions of Jagadish Chandra Bose in Science.

OUTCOME: The Scientific discoveries and inventions of Jagadish Chandra Bose provide inspiration to the readers to make their own contributions to science and technology, and strengthen the nation.

UNIT 5:

1. ' The Health Threats of Climate Change' from English Encounters.

OBJECTIVE:

The essay presents several health disorders that spring out due to environmental changes

OUTCOME:

The lesson offers several inputs to protect environment for the sustainability of the future generations.

2. ' Prafulla Chandra Ray' from The Great Indian Scientists.

OBJECTIVE:

The lesson given an account of the experiments and discoveries in Pharmaceuticals of Prafulla Chandra Ray.

OUTCOME:

Prafulla Chandra Ray's scientific achievements and patriotic fervour provide inspiration to the reader.

UNIT 6:

1. ' The Chief Software Architect' from English Encounters

OBJECTIVE:

The lesson supports the developments of technology for the betterment of human life.

OUTCOME:

Pupil get inspired by eminent personalities who toiled for the present day advancement of software development.

2. ' Srinivasa Ramanujan' from The Great Indian Scientists.

OBJECTIVE:

The lesson highlights the extraordinary achievements of Srinivasa Ramanujan, a great mathematician and the most romantic figure in mathematics.

OUTCOME:

The lesson provides inspiration to the readers to think and tap their innate talents.

NOTE:

All the exercises given in the prescribed lessons in both detailed and non-detailed textbooks relating to the theme and language skills must be covered.

MODEL QUESTION PAPER FOR THEORY

PART- I

Six short answer questions on 6 unit themes

One question on eliciting student's response to any of the themes

PART-II

Each question should be from one unit and the last question can be a combination of two or more units.

Each question should have 3 sub questions: A,B & C

A will be from the main text: 5 marks

B from non-detailed text: 3 marks

C on grammar and Vocabulary: 6 marks

L	T	P	C
4	0	0	3

MATHEMATICS-III
(Common to ALL branches of First Year B.Tech.)

Course Objectives:
1. The course is designed to equip the students with the necessary mathematical skills and techniques that are essential for an engineering course.
2. The skills derived from the course will help the student from a necessary base to develop analytic and design concepts.
3. Understand the most basic numerical methods to solve simultaneous linear equations.

Course Outcomes: At the end of the Course, Student will be able to:
1. Determine rank, Eigenvalues and Eigen vectors of a given matrix and solve simultaneous linear equations.
2. Solve simultaneous linear equations numerically using various matrix methods.
3. Determine double integral over a region and triple integral over a volume.
4. Calculate gradient of a scalar function, divergence and curl of a vector function. Determine line, surface and volume integrals. Apply Green, Stokes and Gauss divergence theorems to calculate line, surface and volume integrals.

UNIT I: Linear systems of equations:

Rank-Echelon form-Normal form – Solution of linear systems – Gauss elimination - Gauss Jordon- Gauss Jacobi and Gauss Seidal methods.Applications: Finding the current in electrical circuits.

UNIT II: Eigen values - Eigen vectors and Quadratic forms:

Eigen values - Eigen vectors– Properties – Cayley-Hamilton theorem - Inverse and powers of a matrix by using Cayley-Hamilton theorem- Diagonalization- Quadratic forms- Reduction of quadratic form to canonical form – Rank - Positive, negative and semi definite - Index – Signature.
Applications: Free vibration of a two-mass system.

UNIT III: Multiple integrals:

Curve tracing: Cartesian, Polar and Parametric forms.
Multiple integrals: Double and triple integrals – Change of variables – Change of order of integration.
Applications: Finding Areas and Volumes.

UNIT IV: Special functions:

Beta and Gamma functions- Properties - Relation between Beta and Gamma functions- Evaluation of improper integrals.
Applications: Evaluation of integrals.

UNIT V: Vector Differentiation:

Gradient- Divergence- Curl - Laplacian and second order operators -Vector identities.
Applications: Equation of continuity, potential surfaces

UNIT VI: Vector Integration:

Line integral – Work done – Potential function – Area- Surface and volume integrals Vector integral theorems: Greens, Stokes and Gauss Divergence theorems (without proof) and related problems.
Applications: Work done, Force.

Text Books:

1. **B.S.Grewal,** Higher Engineering Mathematics, 43rd Edition, Khanna Publishers.
2. **N.P.Bali**, Engineering Mathematics, Lakshmi Publications.

Reference Books:

1. **Greenberg,** Advanced Engineering Mathematics, 2nd edition, Pearson edn
2. **Erwin Kreyszig,** Advanced Engineering Mathematics, 10th Edition, Wiley-India
3. **Peter O'Neil,** Advanced Engineering Mathematics,7th edition, Cengage Learning.
4. **D.W. Jordan and T.Smith,** Mathematical Techniques, Oxford University Press.
5. **Srimanta Pal, Subodh C.Bhunia**, Engineering Mathematics, Oxford University Press.
6. **Dass H.K., Rajnish Verma. Er.,** Higher Engineering Mathematics, S. Chand Co. Pvt. Ltd, Delhi.

	L	T	P	C
	4	0	0	3

I Year - II Semester

APPLIED CHEMISTRY
(EEE, ECE, CSE, IT, EIE, E. Com. E.)

Knowledge of basic concepts of Chemistry for Engineering students will help them as professional engineers later in design and material selection, as well as utilizing the available resources.

Learning Objectives:
- Plastics are nowadays used in household appliances; also they are used as composites (FRP) in aerospace industries.
- Fuels as a source of energy are a basic need of any industry, particularly industries like thermal power stations, steel industry, fertilizer industry etc., and hence they are introduced.
- The basics for the construction of galvanic cells as well as some of the sensors used in instruments are introduced. Also if corrosion is to be controlled, one has to understand the mechanism of corrosion which itself is explained by electrochemical theory.
- With the increase in demand, a wide variety of materials are coming up; some of them have excellent engineering properties and a few of these materials are introduced.
- Understanding of crystal structures will help to understand the conductivity, semiconductors and superconductors. Magnetic properties are also studied.
- With the increase in demand for power and also with depleting sources of fossil fuels, the demand for alternative sources of fuels is increasing. Some of the prospective fuel sources are introduced.

UNIT I: HIGH POLYMERS AND PLASTICS
Polymerisation : Introduction- Mechanism of polymerization - Stereo regular polymers – methods of polymerization (emulsion and suspension) -Physical and mechanical properties – Plastics as engineering materials : advantages and limitations – Thermoplastics and Thermosetting plastics – Compounding and fabrication (4/5 techniques)- Preparation, properties and applications of polyethene, PVC, Bakelite Teflon and polycarbonates
Elastomers – Natural rubber- compounding and vulcanization – Synthetic rubbers : Buna S, Buna N, Thiokol and polyurethanes – Applications of elastomers.
Composite materials & Fiber reinforced plastics – Biodegradable polymers – Conducting polymers.

UNIT II: FUEL TECHNOLOGY
Fuels:- Introduction – Classification – Calorific value - HCV and LCV – Dulong's formula – Bomb calorimeter – Numerical problems – Coal — Proximate and ultimate analysis – Significance of the analyses – Liquid fuels – Petroleum- Refining – Cracking – Synthetic petrol –Petrol knocking – Diesel knocking - Octane and Cetane ratings – Anti-knock agents – Power alcohol – Bio-diesel – Gaseous fuels – Natural gas. LPG and CNG – Combustion – Calculation of air for the combustion of a fuel – Flue gas analysis – Orsat apparatus – Numerical problems on combustion.
Explosives:- Introduction, classification, examples: RDX, TNT and ammonium nitrite - rocket fuels.

UNIT III: ELECTROCHEMICAL CELLS AND CORROSION

Galvanic cells - Reversible and irreversible cells – Single electrode potential – Electro chemical series and uses of this series- Standard electrodes (Hydrogen and Calomel electrodes) - Concentration Cells – Batteries: Dry Cell - Ni-Cd cells - Ni-Metal hydride cells - Li cells - Zinc – air cells.

Corrosion:- Definition – Theories of Corrosion (electrochemical) – Formation of galvanic cells by different metals, by concentration cells, by differential aeration and waterline corrosion – Passivity of metals – Pitting corrosion - Galvanic series – Factors which influence the rate of corrosion - Protection from corrosion – Design and material selection – Cathodic protection - Protective coatings: – Surface preparation – Metallic (cathodic and anodic) coatings - Methods of application on metals (Galvanizing, Tinning, Electroplating, Electroless plating)

UNIT IV: CHEMISTRY OF ADVANCED MATERIALS

Nano materials:- Introduction – Sol-gel method & chemical reduction method of preparation – Characterization by BET method and TEM methods - Carbon nano tubes and fullerenes: Types, preparation, properties and applications

Liquid crystals:- Introduction – Types – Applications

Superconductors :- Type-I & Type-2, properties &applications

Green synthesis:- Principles - 3or 4 methods of synthesis with examples – R_4M_4 principles

UNIT V: SOLID STATE CHEMISTRY

Types of solids - close packing of atoms and ions - BCC , FCC, structures of rock salt - cesium chloride- spinel - normal and inverse spinels,

Non-elemental **semiconducting Materials**:- Stoichiometric, controlled valency & Chalcogen photo/semiconductors, Preparation of Semiconductors / Semiconductor Devices:- p-n junction diode as rectifier – junction transistor.

Insulators (electrical and electronic applications)

Magnetic materials:- Ferro and ferri magnetism. Hall effect and its applications.

UNIT VI: NON CONVENTIONAL ENERGY SOURCES AND STORAGE DEVICES

Solar Energy: - Introduction, application of solar energy, conversion of solar energy (Thermal conversion & photo conversion) – photovoltaic cell: design, working and its importance

Non-conventional energy sources:
  (i)   Hydropower include setup a hydropower plant (schematic diagram)
  (ii)   Geothermal energy: Introduction-schematic diagram of a geothermal power plant
  (iii)   Tidal and wave power: Introduction- Design and working-movement of tides and their effect on sea level.
  (iv)   Ocean thermal energy: Introduction, closed-cycle, ocean thermal energy conversion (OTEC), open cycle OTEC, hybrid OTEC, schematic diagram and explanation.
  (v)   Biomass and biofuels

Fuel cells:- Introduction - cell representation, H_2-O_2 fuel cell: Design and working, advantages and limitations. Types of fuel cells: Alkaline fuel cell - methanol-oxygen - phosphoric acid fuel cells - molten carbonate fuel cells.

Outcomes: The advantages and limitations of plastic materials and their use in design would be understood. Fuels which are used commonly and their economics, advantages and limitations are discussed. Reasons for corrosion and some methods of corrosion control would be understood. The students would be now aware of materials like nano-materials and fullerenes and their uses. Similarly liquid crystals and superconductors are understood. The importance of green synthesis is well understood and how they are different from conventional methods is also explained. Conductance phenomenon is better understood. The students are exposed to some of the alternative fuels and their advantages and limitations.

Standard Books:
1. Engineering Chemistry by Jain and Jain; Dhanpat Rai Publicating Co.
2. Engineering Chemistry by Shikha Agarwal; Cambridge University Press, 2015 edition.

Reference Books:
1. Engineering Chemistry of Wiley India Pvt. Ltd., Vairam and others, 2014 edition (second).
2. Engineering Chemistry by Prasanth Rath, Cengage Learning, 2015 edition.
3. A text book of engineering Chemistry by S. S. Dara, S. Chand & Co Ltd., Latest Edition
4. Applied Chemistry by H.D. Gesser, Springer Publishers
5. Text book of Nano-science and nanotechnology by B.S. Murthy, P. Shankar and others, University Press, IIM

	L	T	P	C
I Year - II Semester	**4**	**0**	**0**	**3**

OBJECT-ORIENTED PROGRAMMING THROUGH C++

OBJECTIVES:
- This course is designed to provide a comprehensive study of the C programming language. It stresses the strengths of C, which provide students with the means of writing efficient, maintainable and portable code. The nature of C language is emphasized in the wide variety of examples and applications. To learn and acquire art of computer programming. To know about some popular programming languages and how to choose
- Programming language for solving a problem.

UNIT-I: Introduction to C++
Difference between C and C++- Evolution of C++- The Object Oriented Technology- Disadvantage of Conventional Programming- Key Concepts of Object Oriented Programming- Advantage of OOP- Object Oriented Language.

UNIT-II: Classes and Objects &Constructors and Destructor
Classes in C++-Declaring Objects- Access Specifiers and their Scope- Defining Member Function-Overloading Member Function- Nested class, Constructors and Destructors, Introduction- Constructors and Destructor- Characteristics of Constructor and Destructor- Application with Constructor- Constructor with Arguments (parameterized Constructor- Destructors- Anonymous Objects.

UNIT-III: Operator Overloading and Type Conversion & Inheritance
The Keyword Operator- Overloading Unary Operator- Operator Return Type- Overloading Assignment Operator (=)- Rules for Overloading Operators, Inheritance, Reusability- Types of Inheritance- Virtual Base Classes- Object as a Class Member- Abstract Classes- Advantages of Inheritance-Disadvantages of Inheritance,

UNIT-IV: Pointers & Binding Polymorphisms and Virtual Functions
Pointer, Features of Pointers- Pointer Declaration- Pointer to Class- Pointer Object- The this Pointer- Pointer to Derived Classes and Base Class, Binding Polymorphisms and Virtual Functions, Introduction- Binding in C++- Virtual Functions- Rules for Virtual Function- Virtual Destructor.

UNIT-V: Generic Programming with Templates & Exception Handling
Generic Programming with Templates, Need for Templates- Definition of class Templates- Normal Function Templates- Over Loading of Template Function-Bubble Sort Using Function Templates- Difference Between Templates and Macros- Linked Lists with Templates, Exception Handling- Principles of Exception Handling- The Keywords try throw and catch- Multiple Catch Statements –Specifying Exceptions.

UNIT-VI: Overview of Standard Template Library
Overview of Standard Template Library- STL Programming Model- Containers- Sequence Containers- Associative Containers- Algorithms- Iterators- Vectors- Lists- Maps.

OUTCOMES:
- Understand the basic terminology used in computer programming
- Write, compile and debug programs in C language. Use different data types in a computer program.
- Design programs involving decision structures, loops and functions.
- Explain the difference between call by value and call by reference

Text Books:
1. A First Book of C++, Gary Bronson, Cengage Learing.
2. The Complete Reference C++, Herbert Schildt, TMH.
3. Programming in C++, Ashok N Kamathane, Pearson 2nd Edition.

Reference Books:
1. Object Oriented Programming C++, Joyce Farrell, Cengage.
2. C++ Programming: from problem analysis to program design, DS Malik, Cengage Learing.

L	T	P	C
4	0	0	3

ENVIRONMENTAL STUDIES

Course Learning Objectives:

The objectives of the course is to impart

- Overall understanding of the natural resources
- Basic understanding of the ecosystem and its diversity
- Acquaintance on various environmental challenges induced due to unplanned anthropogenic activities
- An understanding of the environmental impact of developmental activities
- Awareness on the social issues, environmental legislation and global treaties

Course Outcomes:

The student should have knowledge on

- The natural resources and their importance for the sustenance of the life and recognize the need to conserve the natural resources

- The concepts of the ecosystem and its function in the environment. The need for protecting the producers and consumers in various ecosystems and their role in the food web

- The biodiversity of India and the threats to biodiversity, and conservation practices to protect the biodiversity

- Various attributes of the pollution and their impacts and measures to reduce or control the pollution along with waste management practices

- Social issues both rural and urban environment and the possible means to combat the challenges

- The environmental legislations of India and the first global initiatives towards sustainable development.

- About environmental assessment and the stages involved in EIA and the environmental audit.

- Self Sustaining Green Campus with Environment Friendly aspect of – Energy, Water and Wastewater reuse Plantation, Rain water Harvesting, Parking Curriculum.

Syllabus:

UNIT – I Multidisciplinary nature of Environmental Studies: Definition, Scope and Importance –Sustainability: Stockholm and Rio Summit–Global Environmental Challenges: Global warming and climate change, Carbon Credits, acid rains, ozone layer depletion, population growth and explosion, effects. Role of information Technology in Environment and human health.

Ecosystems: 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 Forest ecosystem, Grassland ecosystem, Desert ecosystem, Aquatic ecosystems.

UNIT – II Natural Resources: Natural resources and associated problems

Forest resources – Use and over – exploitation, deforestation – Timber extraction – Mining, dams and other effects on forest and tribal people

Water resources – Use and over utilization of surface and ground water – Floods, drought, conflicts over water, dams – benefits and problems

Mineral resources: Use and exploitation, environmental effects of extracting and using mineral resources, Sustainable mining of Granite, Literate, Coal, Sea and River sands.

Food resources: World food problems, changes caused by non-agriculture activities-effects of modern agriculture, fertilizer-pesticide problems, water logging, salinity

Energy resources: Growing energy needs, renewable and non-renewable energy sources use of alternate energy sources Vs Oil and Natural Gas Extraction.

Land resources: Land as a resource, land degradation, Wasteland reclamation, man induced landslides, soil erosion and desertification. Role of an individual in conservation of natural resources. Equitable use of resources for sustainable lifestyles.

UNIT – III Biodiversity and its conservation: Definition: genetic, species and ecosystem diversity- classification - Value of biodiversity: consumptive use, productive use, social-Biodiversity at national and local levels. India as a mega-diversity nation - Hot-spots of biodiversity - Threats to biodiversity: habitat loss, man-wildlife conflicts - Endangered and endemic species of India – Conservation of biodiversity: conservation of biodiversity.

UNIT – IV Environmental Pollution: Definition, Cause, effects and control measures of Air pollution, Water pollution, Soil pollution, Noise pollution, Nuclear hazards. Role of an individual in prevention of pollution. - Pollution case studies, Sustainable Life Studies. Impact of Fire Crackers on Men and his well being.

Solid Waste Management: Sources, Classification, effects and control measures of urban and industrial solid wastes. Consumerism and waste products, Biomedical, Hazardous and e – waste management.

UNIT – V Social Issues and the Environment: Urban problems related to energy -Water conservation, rain water harvesting-Resettlement and rehabilitation of people; its problems and concerns. Environmental ethics: Issues and possible solutions. Environmental 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 Environmental Management: Impact Assessment and its significance various stages of EIA, preparation of EMP and EIS, Environmental audit. Ecotourism, Green Campus – Green business and Green politics.

The student should Visit an Industry / Ecosystem and submit a report individually on any issues related to Environmental Studies course and make a power point presentation.

Text Books:

1. Environmental Studies, K. V. S. G. Murali Krishna, VGS Publishers, Vijayawada

2. Environmental Studies, R. Rajagopalan, 2nd Edition, 2011, Oxford University Press.

3. Environmental Studies, P. N. Palanisamy, P. Manikandan, A. Geetha, and K. Manjula Rani; Pearson Education, Chennai

Reference:

1. Text Book of Environmental Studies, Deeshita Dave & P. Udaya Bhaskar, Cengage Learning.

2. A Textbook of Environmental Studies, Shaashi Chawla, TMH, New Delhi

3. Environmental Studies, Benny Joseph, Tata McGraw Hill Co, New Delhi

4. Perspectives in Environment Studies, Anubha Kaushik, C P Kaushik, New Age International Publishers, 2014

L	T	P	C
4	0	0	3

ENIGINEERING MECHANICS

Objectives: The students completing this course are expected to understand the concepts of forces and its resolution in different planes, resultant of force system, Forces acting on a body, their free body diagrams using graphical methods. They are required to understand the concepts of centre of gravity and moments of inertia and their application, Analysis of frames and trusses, different types of motion, friction and application of work - energy method.

UNIT – I
Objectives: The students are to be exposed to the concepts of force and friction, direction and its application.
Introduction to Engg. Mechanics – Basic Concepts.

Systems of Forces: Coplanar Concurrent Forces – Components in Space – Resultant – Moment of Force and its Application – Couples and Resultant of Force Systems.

Friction: Introduction, limiting friction and impending motion, coulomb's laws of dry friction, coefficient of friction, cone of friction

UNIT II
Objectives: The students are to be exposed to application of free body diagrams. Solution to problems using graphical methods and law of triangle of forces.

Equilibrium of Systems of Forces: Free Body Diagrams, Equations of Equilibrium of Coplanar Systems, Spatial Systems for concurrent forces. Lamis Theorm, Graphical method for the equilibrium of coplanar forces, Converse of the law of Triangle of forces, converse of the law of polygon of forces condition of equilibrium, analysis of plane trusses.

UNIT – III
Objectives : The students are to be exposed to concepts of centre of gravity.
Centroid: Centroids of simple figures (from basic principles) – Centroids of Composite Figures

Centre of Gravity: Centre of gravity of simple body (from basic principles), centre of gravity of composite bodies, Pappus theorems.

UNIT IV
Objective: The students are to be exposed to concepts of moment of inertia and polar moment of inertia including transfer methods and their applications.
Area moments of Inertia: Definition – Polar Moment of Inertia, Transfer Theorem, Moments of Inertia of Composite Figures, Products of Inertia, Transfer Formula for Product of Inertia.
Mass Moment of Inertia: Moment of Inertia of Masses, Transfer Formula for Mass Moments of Inertia, mass moment of inertia of composite bodies.

UNIT – V

Objectives: The students are to be exposed to motion in straight line and in curvilinear paths, its velocity and acceleration computation and methods of representing plane motion.
Kinematics: Rectilinear and Curvelinear motions – Velocity and Acceleration – Motion of Rigid Body – Types and their Analysis in Planar Motion. **Kinetics:** Analysis as a Particle and Analysis as a Rigid Body in Translation – Central Force Motion – Equations of Plane Motion – Fixed Axis Rotation – Rolling Bodies.

UNIT – VI

Objectives: The students are to be exposed to concepts of work, energy and particle motion
Work – Energy Method: Equations for Translation, Work-Energy Applications to Particle Motion, Connected System-Fixed Axis Rotation and Plane Motion. Impulse momentum method.

Text Books :

1. Engg. Mechanics - S.Timoshenko & D.H.Young., 4th Edn - , Mc Graw Hill publications.

References:

1. Engineering Mechanics statics and dynamics – R.C.Hibbeler, 11th Edn – Pearson Publ.
2. Engineering Mechanics, statics – J.L.Meriam, 6th Edn – Wiley India Pvt Ltd.
3. Engineering Mechanics, statics and dynamics – I.H.Shames, – Pearson Publ.
4. Mechanics For Engineers, statics - F.P.Beer & E.R.Johnston – 5th Edn Mc Graw Hill Publ.
5. Mechanics For Engineers, dynamics - F.P.Beer & E.R.Johnston –5th Edn Mc Graw Hill Publ.
6. Theory & Problems of engineering mechanics, statics & dynamics – E.W.Nelson, C.L.Best & W.G. McLean, 5th Edn – Schaum's outline series - Mc Graw Hill Publ.
7. Singer's Engineering Mechanics: Statics And Dynamics, K. Vijay Kumar Reddy, J. Suresh Kumar, Bs Publications
8. Engineering Mechanics, Fedinand . L. Singer, Harper – Collins.
9. Engineering Mechanics statics and dynamics , A Nelson , Mc Graw Hill publications

L	T	P	C
0	0	3	2

I Year - II Semester

APPLIED / ENGINEERING CHEMISTRY LABORATORY (*Common to all branches*)

1. Introduction to Chemistry laboratory – Molarity, Normality, Primary, secondary standard solutions, Volumetric titrations, Quantitative analysis, Qualitative analysis, etc.

2. Trial experiment - Determination of HCl using standard Na_2CO_3 solution.

3. Determination of alkalinity of a sample containing Na_2CO_3 and NaOH.

4. Determination of $KMnO_4$ using standard Oxalic acid solution.

5. Determination of Ferrous iron using standard $K_2Cr_2O_7$ solution.

6. Determination of Copper using standard $K_2Cr_2O_7$ solution.

7. Determination of temporary and permanent hardness of water using standard EDTA solution.

8. Determination of Copper using standard EDTA solution.

9. Determination of Iron by a Colorimetric method using thiocynate as reagent.

10. Determination of pH of the given sample solution using pH meter.

11. Conductometric titration between strong acid and strong base.

12. Conductometric titration between strong acid and weak base.

13. Potentiometric titration between strong acid and strong base.

14. Potentiometric titration between strong acid and weak base.

15. Determination of Zinc using standard EDTA solution.

16. Determination of Vitamin – C.

Outcomes: The students entering into the professional course have practically very little exposure to lab classes. The experiments introduce volumetric analysis; redox titrations with different indicators; EDTA titrations; then they are exposed to a few instrumental methods of chemical analysis. Thus at the end of the lab course, the student is exposed to different methods of chemical analysis and use of some commonly employed instruments. They thus acquire some experimental skills.

Reference Books

1. A Textbook of Quantitative Analysis, Arthur J. Vogel.
2. Dr. Jyotsna Cherukuris (2012) *Laboratory Manual of engineering chemistry-II,* VGS Techno Series
3. Chemistry Practical Manual, Lorven Publications
4. K. Mukkanti (2009) *Practical Engineering Chemistry,* B.S. Publication

I Year - II Semester	ENGLISH - COMMUNICATION SKILLS LAB- II	L	T	P	C
		0	0	3	2

PRESCRIBED LAB MANUAL FOR SEMESTER II:

'INTERACT: English Lab Manual for Undergraduate Students' Published by **Orient Blackswan Pvt Ltd.**

OBJECTIVES:

To enable the students to learn demonstratively the communication skills of listening, speaking, reading and writing.

OUTCOME:

A study of the communicative items in the laboratory will help the students become successful in the competitive world.

The course content along with the study material is divided into six units.

UNIT 1:

1. Debating
   Practice work

UNIT 2:

1. Group Discussions
   Practice work

UNIT 3:

1. Presentation Skills
   Practice work

UNIT 4:

1. Interview Skills
   Practice work

UNIT 5:

1. Email,
2. Curriculum Vitae
   Practice work

UNIT 6:

1. Idiomatic Expressions
2. Common Errors in English
   Practice work

Reference Books:

1. Strengthen your communication skills by Dr M Hari Prasad, Dr Salivendra Raju and Dr G Suvarna Lakshmi, Maruti Publications.
2. English for Professionals by Prof Eliah, B.S Publications, Hyderabad.
3. Unlock, Listening and speaking skills 2, Cambridge University Press
4. Spring Board to Success, Orient BlackSwan
5. A Practical Course in effective english speaking skills, PHI
6. Word power made handy, Dr shalini verma, Schand Company
7. Let us hear them speak, Jayashree Mohanraj, Sage texts
8. Professional Communication, Aruna Koneru, Mc Grawhill Education
9. Cornerstone, Developing soft skills, Pearson Education

L	T	P	C
0	0	3	2

OBJECT-ORIENTED PROGRAMMING LAB

OBJECTIVE

- To strengthen their problem solving ability by applying the characteristics of an object- oriented approach.

- To introduce object oriented concepts in C++ and Java.

Programmig:

Exercise – 1 (Basics)
Write a Simple Program on printing "Hello World" and "Hello Name" where name is the input from the user
- a) Convert any two programs that are written in C into C++
- b) Write a description of using g++ (150 Words)

Exercise – 2 (Expressions Control Flow)
a) Write a Program that computes the simple interest and compound interest payable on principalamount(inRs.)ofloanborrowedbythecustomerfromabankforagiverperiodof time (in years) at specific rate of interest. Further determine whether the bank will benefit by charging simple interest or compound interest.

b) WriteaProgramtocalculatethefareforthepassengerstravelinginabus. WhenaPassenger enters the bus, the conductor asks "What distance will you travel?" On knowing distance from passenger (as an approximate integer), the conductor mentions the fare to the passenger according to following criteria.

Exercise – 3 (Variables, Scope, Allocation)

a) Write a program to implement call by value and call by reference using reference variable.

b) Write a program to illustrate scope resolution, new and delete Operators. (Dyanamic Memory
Allocation)

c) Write a program to illustrate Storage classes

d) Write a program to illustrate Enumerations

Exercises –4 (Functions)

Write a program illustrating Inline Functions

    a) Write a program illustrate function overloading. Write 2 overloading functions for power.

    b) Write a program illustrate the use of default arguments for simple interest function.

Exercise -5 (Functions –Exercise   Continued)

a) Write a program to illustrate function overloading. Write 2 overloading functions for adding two numbers

b) Write a program illustrate function template for power of a number.

c) Write a program to illustrate function template for swapping of two numbers.

Exercise -6 (Classes Objects)

Create a Distance class with:

    • feet and inches as data members
    • member function to input distance
    • member function to output distance
    • member function to add two distance objects

a). Write a main function to create objects of DISTANCE class. Input two distances and output the sum.

b). Write a C++ Program to illustrate the use of Constructors and Destructors (use the above program.)

c) Write a program for illustrating function overloading in adding the distance between objects (use the above problem)

d). Write a C++ program demonstrating a BankAccount with necessary methods and variables

Exercise – 7 (Access)

Write a program for illustratingAccess Specifiers public, private, protected

a) Write a program implementing Friend Function
b) Write a program to illustrate this pointer
c) Write a Program to illustrate pointer to a class
d)

Exercise -8 (Operator Overloading)

a). Write a program to Overload Unary, and Binary Operators as Member Function, and Non Member Function.

i. Unary operator as member function

ii. Binary operator as nonmember function

b). Write a c ++ program to implement the overloading assignment = operator

c).Write a case study on Overloading Operators and Overloading Functions (150 Words)

Exercise -9 (Inheritance)
 a) Write C++ Programs and incorporating various forms of Inheritance
   i)   Single Inheritance
   ii) Hierarchical Inheritance
   iii) Multiple Inheritances
   iv) Multi-level inheritance
   v) Hybrid inheritance
b) Write a program to show Virtual Base Class

c) Write a case study on using virtual classes (150 Words)

Exercise-10 (Inheritance –Continued)
a) Write a Program in C++ to illustrate the order of execution of constructors and destructors in inheritance
b) Write a Program to *show* how *constructors* are invoked in *derived class*

Exercise -11 (Polymorphism)
   a) Write a program to illustrate runtime polymorphism
   b) Write a program to illustrate this pointer
   c) Write a program illustrates pure virtual function and calculate the area of different shapes by using abstract class.
   d) Write a case study on virtual functions (150 Words)

Exercise -12(Templates)
   a) Write a C++ Program to illustrate template class

   b) Write a Program to illustrate class templates with multiple parameters

   c) Write a Program to illustrate member function templates

Exercise -13 (Exception Handling)
a).Write a Program for Exception Handling Divide by zero

 b). Write a Program to rethrow an Exception

Exercise -14 (STL)
a) Write a Program to implement List and List Operations

b) Write a Program to implementVector andVector Operations

Exercise -15 (STLContinued)
a) Write a Program to implement Deque and Deque Operations

b) Write a Program to implement Map and Map Operations

OUTCOMES:

- Explain what constitutes an object-oriented approach to programming and identify potential benefits of object-oriented programming over other approaches.

- Apply an object-oriented approach to developing applications of varying complexities

II Year - I Semester

L	T	P	C
4	0	0	3

STATISTICS WITH R PROGRAMMING

OBJECTIVE:

After taking the course, students will be able to

- Use R for statistical programming, computation, graphics, and modeling,
- Write functions and use R in an efficient way,
- Fit some basic types of statistical models
- Use R in their own research,
- Be able to expand their knowledge of R on their own.

UNIT-I:

Introduction, How to run R, R Sessions and Functions, Basic Math, Variables, Data Types, Vectors, Conclusion, Advanced Data Structures, Data Frames, Lists, Matrices, Arrays, Classes**.**

UNIT-II:

R Programming Structures, Control Statements, Loops, - Looping Over Nonvector Sets,- If-Else, Arithmetic and Boolean Operators and values, Default Values for Argument, Return Values, Deciding Whether to explicitly call return- Returning Complex Objects, Functions are Objective, No Pointers in R, Recursion, A Quicksort Implementation-Extended Extended Example: A Binary Search Tree.

UNIT-III:

Doing Math and Simulation in R, Math Function, Extended Example Calculating Probability- Cumulative Sums and Products-Minima and Maxima- Calculus, Functions Fir Statistical Distribution, Sorting, Linear Algebra Operation on Vectors and Matrices, Extended Example: Vector cross Product- Extended Example: Finding Stationary Distribution of Markov Chains, Set Operation, Input /out put, Accessing the Keyboard and Monitor, Reading and writer Files,

UNIT-IV:

Graphics, Creating Graphs, The Workhorse of R Base Graphics, the plot() Function – Customizing Graphs, Saving Graphs to Files.

UNIT-V:

Probability Distributions, Normal Distribution- Binomial Distribution- Poisson Distributions Other Distribution, Basic Statistics, Correlation and Covariance, T-Tests,-ANOVA.

UNIT-VI:

Linear Models, Simple Linear Regression, -Multiple Regression Generalized Linear Models, Logistic Regression, - Poisson Regression- other Generalized Linear Models-Survival Analysis, Nonlinear Models, Splines- Decision- Random Forests,

OUTCOMES:

At the end of this course, students will be able to:

• List motivation for learning a programming language

• Access online resources for R and import new function packages into the R workspace

• Import, review, manipulate and summarize data-sets in R

• Explore data-sets to create testable hypotheses and identify appropriate statistical tests

• Perform appropriate statistical tests using R Create and edit visualizations with

TEXT BOOKS:

1) The Art of R Programming, Norman Matloff, Cengage Learning
2) R for Everyone, Lander, Pearson

REFERENCE BOOKS:

1) R Cookbook, PaulTeetor, Oreilly.
2) R in Action,Rob Kabacoff, Manning

	L	T	P	C
II Year - I Semester

	L	T	P	C
	4	0	0	3

MATHEMATICAL FOUNDATION OF COMPUTER SCIENCE

OBJECTIVES:
- To introduce the students to the topics and techniques of discrete methods and combinatorial reasoning.
- To introduce a wide variety of applications. The algorithmic approach to the solution of problems is fundamental in discrete mathematics, and this approach reinforces the close ties between this discipline and the area of computer science.

UNIT -I:
Mathematical Logic: Propositional Calculus: Statements and Notations, Connectives, Well Formed Formulas, Truth Tables, Tautologies, Equivalence of Formulas, Duality Law, Tautological Implications, Normal Forms, Theory of Inference for Statement Calculus, Consistency of Premises, Indirect Method of Proof, Predicate Calculus:Predicative Logic, Statement Functions, Variables and Quantifiers, Free and Bound Variables, Inference Theory for Predicate Calculus.

UNIT -II:
Set Theory: Introduction, Operations on Binary Sets, Principle of Inclusion and Exclusion, *Relations:* Properties of Binary Relations, Relation Matrix and Digraph, Operations on Relations, Partition and Covering, Transitive Closure, Equivalence, Compatibility and Partial Ordering Relations, Hasse Diagrams, *Functions:* Bijective Functions, Composition of Functions, Inverse Functions, Permutation Functions, Recursive Functions, Lattice and its Properties.

UNIT- III:
Algebraic Structures and Number Theory: *Algebraic Structures:*Algebraic Systems, Examples, General Properties, Semi Groups and Monoids, Homomorphism of Semi Groups and Monoids, Group, Subgroup, Abelian Group, Homomorphism, Isomorphism, *Number Theory:*Properties of Integers, Division Theorem, The Greatest Common Divisor, Euclidean Algorithm, Least Common Multiple, Testing for Prime Numbers, The Fundamental Theorem of Arithmetic, Modular Arithmetic (Fermat's Theorem and Euler's Theorem)

UNIT -IV:
Combinatorics: Basic of Counting, Permutations, Permutations with Repetitions, Circular Permutations, Restricted Permutations, Combinations, Restricted Combinations, Generating Functions of Permutations and Combinations, Binomial and Multinomial Coefficients, Binomial and Multinomial Theorems, The Principles of Inclusion–Exclusion, Pigeonhole Principle and its Application.

UNIT -V:

Recurrence Relations: Generating Functions, Function of Sequences, Partial Fractions, Calculating Coefficient of Generating Functions, Recurrence Relations, Formulation as Recurrence Relations, Solving Recurrence Relations by Substitution and Generating Functions, Method of Characteristic Roots, Solving Inhomogeneous Recurrence Relations

UNIT -VI:

Graph Theory: Basic Concepts of Graphs, Sub graphs, Matrix Representation of Graphs: Adjacency Matrices, Incidence Matrices, Isomorphic Graphs, Paths and Circuits, Eulerian and Hamiltonian Graphs, Multigraphs, Planar Graphs, Euler's Formula, Graph Colouring and Covering, Chromatic Number, Spanning Trees, Algorithms for Spanning Trees (Problems Only and Theorems without Proofs).

OUTCOMES:
- Student will be able to demonstrate skills in solving mathematical problems
- Student will be able to comprehend mathematical principles and logic
- Student will be able to demonstrate knowledge of mathematical modeling and proficiency in using mathematical software
- Student will be able to manipulate and analyze data numerically and/or graphically using appropriate Software
- Student will be able to communicate effectively mathematical ideas/results verbally or in writing

TEXT BOOKS:

1.Discrete Mathematical Structures with Applications to Computer Science, J. P. Tremblay and P. Manohar, Tata McGraw Hill.

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

3. Discrete Mathematics and its Applications with Combinatorics and Graph Theory, K. H. Rosen, 7th Edition, Tata McGraw Hill.

REFERENCE BOOKS:

1. Discrete Mathematics for Computer Scientists and Mathematicians, J. L. Mott, A. Kandel, T.P. Baker, 2nd Edition, Prentice Hall of India.

2. Discrete Mathematical Structures, BernandKolman, Robert C. Busby, Sharon Cutler Ross, PHI.

3. Discrete Mathematics, S. K. Chakraborthy and B.K. Sarkar, Oxford, 2011.

II Year - I Semester

L	T	P	C
4	0	0	3

DIGITAL LOGIC DESIGN

OBJECTIVE:
- To introduce the basic tools for design with combinational and sequential digital logic and state machines.
- To learn simple digital circuits in preparation for computer engineering.

UNIT- I: Digital Systems and Binary Numbers

Digital Systems, Binary Numbers, Binary Numbers, Octal and Hexadecimal Numbers, Complements of Numbers, Complements of Numbers, Signed Binary Numbers,Arithmetic addition and subtraction

UNIT -II: Concept of Boolean algebra

Basic Theorems and Properties of Boolean algebra, Boolean Functions, Canonical and Standard Forms, Minterms and Maxterms,

UNIT- III: Gate level Minimization

Map Method, Two-Variable K-Map, Three-Variable K-Map, Four Variable K-Maps. Products of Sum Simplification, Sum of Products Simplification, Don't – Care Conditions, NAND and NOR Implementation, Exclusive-OR Function

UNIT- IV:Combinational Logic

Introduction, Analysis Procedure, Design Procedure, Binary Adder–Subtractor, Decimal Adder, Binary Multiplier, Decoders, Encoders, Multiplexers, HDL Models of Combinational Circuits

UNIT- V: Synchronous Sequential Logic

Introduction to Sequential Circuits, Storage Elements: Latches, Storage Elements: Flip-Flops, Analysis of Clocked **Sequential** Circuits, Mealy and Moore Models of Finite State Machines

UNIT -VI: Registers and Counters

Registers, Shift Registers, Ripple Counters, Synchronous Counters, Ring Counter, Johnson Counter, Ripple Counter

OUTCOMES:

A student who successfully fulfills the course requirements will have demonstrated:

- An ability to define different number systems, binary addition and subtraction, 2's complement representation and operations with this representation.
- An ability to understand the different switching algebra theorems and apply them for logic functions.
- An ability to define the Karnaugh map for a few variables and perform an algorithmic reduction of logic functions.
- An ability to define the other minimization methods for any number of variables Variable Entered Mapping (VEM) and Quine-MeCluskey (QM) Techniques and perform an algorithmic reduction of logic functions.

TEXT BOOKS:

1. Digital Design, 5/e, M.Morris Mano, Michael D Ciletti, PEA.
2. Fundamentals of Logic Design, 5/e, Roth, Cengage.

REFERENCE BOOKS:

1. Digital Logic and Computer Design, M.Morris Mano, PEA.
2. Digital Logic Design, Leach, Malvino, Saha, TMH.
3. Modern Digital Electronics, R.P. Jain, TMH.

L	T	P	C
4	0	0	3

PYTHON PROGRAMMING

OBJECTIVES:

- Introduction to Scripting Language
- Exposure to various problems solving approaches of computer science

UNIT – I:
Introduction: History of Python, Need of Python Programming, Applications Basics of Python Programming Using the REPL(Shell), Running Python Scripts, Variables, Assignment, Keywords, Input-Output, Indentation.

UNIT – II:
Types, Operators and Expressions: Types - Integers, Strings, Booleans; Operators- Arithmetic Operators, Comparison (Relational) Operators, Assignment Operators, Logical Operators, Bitwise Operators, Membership Operators, Identity Operators, Expressions and order of evaluations Control Flow- if, if-elif-else, for, while, break, continue, pass

UNIT – III:
Data Structures Lists - Operations, Slicing, Methods; Tuples, Sets, Dictionaries, Sequences. Comprehensions.

UNIT – IV:
Functions - Defining Functions, Calling Functions, Passing Arguments, Keyword Arguments, Default Arguments, Variable-length arguments, Anonymous Functions, Fruitful Functions(Function Returning Values), Scope of the Variables in a Function - Global and Local Variables.

Modules: Creating modules, import statement, from. Import statement, name spacing,

Python packages, Introduction to PIP, Installing Packages via PIP, Using Python Packages

UNIT – V:
Object Oriented Programming OOP in Python: Classes, 'self variable', Methods, Constructor Method, Inheritance, Overriding Methods, Datahiding,

Error and Exceptions: Difference between an error and Exception, Handling Exception, try except block, Raising Exceptions, User Defined Exceptions

UNIT – VI:

Brief Tour of the Standard Library - Operating System Interface - String Pattern Matching, Mathematics, Internet Access, Dates and Times, Data Compression, Multithreading, GUI Programming, Turtle Graphics

Testing: Why testing is required ?, Basic concepts of testing, Unit testing in Python, Writing Test cases, Running Tests.

OUTCOMES:
- Making Software easily right out of the box.
- Experience with an interpreted Language.
- To build software for real needs.
- Prior Introduction to testing software

TEXT BOOKS
1. Python Programming: A Modern Approach, Vamsi Kurama, Pearson
2. Learning Python, Mark Lutz, Orielly

Reference Books:
1. Think Python, Allen Downey, Green Tea Press
2. Core Python Programming, W.Chun, Pearson.
3. Introduction to Python, Kenneth A. Lambert, Cengage

	L	T	P	C
II Year - I Semester | 4 | 0 | 0 | 3 |

DATA STRUCTURES THROUGH C++

OBJECTIVES:

- To be familiar with basic techniques of object oriented principles and exception handling using C++
- To be familiar with the concepts like Inheritance, Polymorphism
- Solve problems using data structures such as linear lists, stacks, queues, hash tables
- Be familiar with advanced data structures such as balanced search trees, AVLTrees, and B Trees.

UNIT-I: ARRAYS

Abstract Data Types and the C++ Class, An Introduction to C++ Class- Data Abstraction and Encapsulation in C++- Declaring Class Objects and Invoking Member Functions- Special Class Operations- Miscellaneous Topics- ADTs and C++Classes, The Array as an Abstract Data Type, The Polynomial Abstract Data type-   Polynomial Representation- Polynomial Addition. Spares Matrices,Introduction- Sparse Matrix Representation- Transposing a Matrix- Matrix Multiplication, Representation of Arrays.

UNIT-II: STACKS AND QUEUES

Templates in C++, Template Functions- Using Templates to Represent Container Classes, The Stack Abstract Data Type, The Queue Abstract Data Type, Subtyping and Inheritance in C++, Evaluation of Expressions, Expression- Postfix Notation- Infix to Postfix.

UNIT-III: LINKED LISTS

Single Linked List and Chains, Representing Chains in C++, Defining a Node in C++- Designing a Chain Class in C++- Pointer manipulation in C++- Chain Manipulation Operations, The Template Class Chain, Implementing Chains with Templates- Chain Iterators- Chain Operations- Reusing a Class, Circular Lists, Available Space Lists, Linked Stacks and Queues, Polynomials, Polynomial Representation- Adding Polynomials- Circular List Representation of Polynomials, Equivalence Classes, Sparse Matrices, Sparse Matrix Representation- Sparse Matrix Input- Deleting a Sparse Matrix, Doubly Linked Lists, Generalized Lists, Representation of Generalized Lists- Recursive Algorithms for Lists- Reference Counts, Shared and Recursive Lists

UNIT-IV: TREES

Introduction, Terminology, Representation of Trees, Binary Trees, The Abstract Data Type, Properties of Binary Tress, Binary Tree Representations, Binary Tree Traversal and Tree Iterators, Introduction, Inorder Traversal Preorder Traversal, Postorder Traversal, Thread Binary Trees, Threads, Inorder Traversal of a Threaded Binary Tree, Inserting a Node into a Threaded Binary Tree, Heaps, Priority Queues, Definition of a Max Heap, Insertion into a Max Heap, Deletion from a Max Heap, Binary Search Trees, Definition, Searching a Binary Search Tree, Insertion into a Binary Search Tree, Deletion from a Binary Search Tree, Height of Binary Search Tree.

UNIT-V: GRAPHS

The Graph Abstract Data Type, Introduction, Definition, Graph Representation, Elementary Graph Operation, Depth First Search, Breadth First Search, Connected Components, Spanning Trees, Biconnected Components, Minimum Cost Spanning Trees, Kruskal S Algorithm, Prim s Algorithm Sollin' s Algorithm, Shortest Paths and Transitive Closure, Single Source/All Destination: Nonnegative Edge Cost, Single Source/All Destination: General Weights, All-Pairs Shortest Path, Transitive Closure.

UNIT-VI: SORTING

Insertion Sort, Quick Sort, Merge Sort Merging, Iterative Merge Sort, Recursive Merge Sort, Heap Sort.

OUTCOMES:

- Distinguish between procedures and object oriented programming.
- Apply advanced data structure strategies for exploring complex data structures.
- Compare and contrast various data structures and design techniques in the area of Performance.
- Implement data structure algorithms through C++. • Incorporate data structures into the applications such as binary search trees, AVL and B Trees
- Implement all data structures like stacks, queues, trees, lists and graphs and compare their Performance   and trade offs

TEXT BOOKS:

1. Data structures, Algorithms and Applications in C++, S.Sahni, University Press (India) Pvt.Ltd, 2nd edition, Universities Press, Pvt. Ltd.
2. Data structures and Algorithm Analysis in C++, Mark Allen Weiss, Pearson Education. Ltd., Second Edition.
3. Data structures and Algorithms in C++, Michael T.Goodrich, R.Tamassia and .Mount, Wiley student edition, John Wiley and Sons.

REFERENCE BOOKS:

1. Data structures and algorithms in C++, 3rd Edition, Adam Drozdek, Thomson
2. Data structures using C and C++, Langsam, Augenstein and Tanenbaum, PHI.
3. Problem solving with C++, The OOP, Fourth edition, W.Savitch, Pearson education.

II Year - I Semester

L	T	P	C
4	0	0	3

COMPUTER GRAPHICS

OBJECTIVES:

- To develop, design and implement two and three dimensional graphical structures
- To enable students to acquire knowledge Multimedia compression and animations
- To learn Creation, Management and Transmission of Multimedia objects.

UNIT-I:
2D Primitives Output primitives – Line, Circle and Ellipse drawing algorithms - Attributes of output primitives – Two dimensional Geometric transformations - Two dimensional viewing – Line, Polygon, Curve and Text clipping algorithms

UNIT-II:
3D Concepts Parallel and Perspective projections - Three dimensional object representation –

Polygons, Curved lines, Splines, Quadric Surfaces, - Visualization of data sets -

3Dtransformations – Viewing -Visible surface identification.

UNIT-III:
Graphics Programming Color Models – RGB, YIQ, CMY, HSV – Animations – General

Computer Animation, Raster, Keyframe - Graphics programming using OPENGL – Basic

graphics primitives –Drawing three dimensional objects - Drawing three dimensional scenes

UNIT- IV:
Rendering Introduction to Shading models – Flat and Smooth shading – Adding texture to faces –Adding shadows of objects – Building a camera in a program – Creating shaded objects– Rendering texture – Drawing Shadows.

UNIT- V:
FractalsFractals and Self similarity – Peano curves – Creating image by iterated functions – Mandelbrot sets – Julia Sets – Random Fractals

UNIT- VI:

Overview of Ray Tracing Intersecting rays with other primitives – Adding Surface texture –

Reflections andTransparency – Boolean operations on Objects.

OUTCOMES:

- Know and be able to describe the general software architecture of programs that use 3D computer graphics.

- Know and be able to discuss hardware system architecture for computer graphics. This Includes, but is not limited to: graphics pipeline, frame buffers, and graphic accelerators/co-processors.

- Know and be able to select among models for lighting/shading: Color, ambient light; distant and light with sources; Phong reflection model; and shading (flat, smooth, Gourand, Phong).

TEXT BOOKS:

1. Donald Hearn, Pauline Baker, Computer Graphics – C Version, second edition   Pearson Education,2004.

2. F.S. Hill, Computer Graphics using OPENGL, Second edition, Pearson Education, 2003.

REFERENCE BOOKS:

1. James D. Foley, Andries Van Dam, Steven K. Feiner, John F. Hughes, Computer Graphics- Principles and practice, Second Edition in C, Pearson Education, 2007.

L	**T**	**P**	**C**
0	0	3	2

II Year - I Semester

DATASTRUCTURES THROUGH C++ LAB

OBJECTIVES:

- To develop skills to design and analyze simple linear and non linear data structures

- To Strengthen the ability to identify and apply the suitable data structure for the given real world problem

- To Gain knowledge in practical applications of data structures

List of Experiments:

1. Implementation of Singly linked list.

2. Implementation of Doubly linked list.

3. Implementation of Multistack in a Single Array.

4. Implementation of Circular Queue

5. Implementation of Binary Search trees.

6. Implementation of Hash table.

7. Implementation of Heaps.

8. Implementation of Breadth First Search Techniques.

9. Implementation of Depth First Search Techniques.

10. Implementation of Prim's Algorithm.

11. Implementation of Dijkstra's Algorithm.

12. Implementation of Kruskal's Algorithm

13. Implementation of MergeSort

14. Implementation of Quick Sort

15. Implementation of Data Searching using divide and conquer technique

OUTCOMES:

At the end of this lab session, the student will

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

- Be capable to identity the appropriate data structure for given problem

Have practical knowledge on the application of data structures

		L	T	P	C
II Year - I Semester		**0**	**0**	**3**	**2**

PYTHON PROGRAMMING LAB

Exercise 1 - Basics

a)   Running instructions in Interactive interpreter and a Python Script
b)   Write a program to purposefully raise Indentation Error and Correct it

Exercise 2 - Operations

a)   Write a program to compute distance between two points taking input from the user (Pythagorean Theorem)
b) Write a program add.py that takes 2 numbers as command line arguments and prints its sum.

Exercise - 3 Control Flow

a) Write a Program for checking whether the given number is a even number or not.

b) Using a for loop, write a program that prints out the decimal equivalents of 1/2, 1/3, 1/4, . . . , 1/10

c) Write a program using a for loop that loops over a sequence. What is sequence ?

d) Write a program using a while loop that asks the user for a number, and prints a countdown from that number to zero.

Exercise 4 - Control Flow - Continued

a) Find the sum of all the primes below two million.
   Each new term in the Fibonacci sequence is generated by adding the previous two terms. By starting with 1 and 2, the first 10 terms will be:

1, 2, 3, 5, 8, 13, 21, 34, 55, 89, ...

b) By considering the terms in the Fibonacci sequence whose values do not exceed four million, find the sum of the even-valued terms.

Exercise - 5 - DS

a)   Write a program to count the numbers of characters in the string and store them in a dictionary data structure
b)   Write a program to use split and join methods in the string and trace a birthday with a dictionary data structure.

Exercise - 6 DS - Continued

a) Write a program combine_lists that combines these lists into a dictionary.

b) Write a program to count frequency of characters in a given file. Can you use character frequency to tell whether the given file is a Python program file, C program file or a text file?

Exercise - 7 Files

a) Write a program to print each line of a file in reverse order.
b) Write a program to compute the number of characters, words and lines in a file.

Exercise - 8 Functions

a) Write a function ball_collide that takes two balls as parameters and computes if they are colliding. Your function should return a Boolean representing whether or not the balls are colliding.

   Hint: Represent a ball on a plane as a tuple of (x, y, r), r being the radius

   If (distance between two balls centers) <= (sum of their radii) then (they are colliding)

b) Find mean, median, mode for the given set of numbers in a list.

Exercise - 9 Functions - Continued

a) Write a function nearly_equal to test whether two strings are nearly equal. Two strings a and b are nearly equal when a can be generated by a single mutation on b.
b) Write a function dups to find all duplicates in the list.
c) Write a function unique to find all the unique elements of a list.

Exercise - 10 - Functions - Problem Solving

a) Write a function cumulative_product to compute cumulative product of a list of numbers.
b) Write a function reverse to reverse a list. Without using the reverse function.
c) Write function to compute gcd, lcm of two numbers. Each function shouldn't exceed one line.

Exercise 11 - Multi-D Lists

a) Write a program that defines a matrix and prints
b) Write a program to perform addition of two square matrices
c) Write a program to perform multiplication of two square matrices

Exercise - 12 - Modules

a) Install packages requests, flask and explore them. using (pip)
b) Write a script that imports requests and fetch content from the page. Eg. (Wiki)
c) Write a simple script that serves a simple HTTPResponse and a simple HTML Page

Exercise - 13 OOP

a) Class variables and instance variable and illustration of the self variable
   i) Robot
   ii) ATM Machine

Exercise - 14 GUI, Graphics

1. Write a GUI for an Expression Calculator using tk
2. Write a program to implement the following figures using turtle
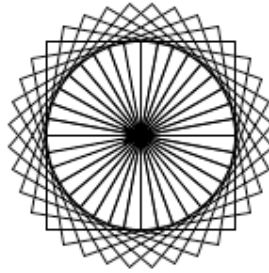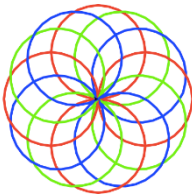
Exercise - 15 - Testing

a) Write a test-case to check the function even_numbers which return True on passing a list of all even numbers
b) Write a test-case to check the function reverse_string which returns the reversed string

Exercise - 16 - Advanced

a)   Build any one classical data structure.
b)   Write a program to solve knapsack problem.

	L	T	P	C
II Year – II Semester	4	0	0	3

SOFTWARE ENGINEERING

OBJECTIVES

- To understand the software life cycle models.
- To understand the software requirements and SRS document.
- To understand the importance of modeling and modeling languages.
- To design and develop correct and robust software products.
- To understand the quality control and how to ensure good quality software.
- To understand the planning and estimation of software projects.
- To understand the implementation issues, validation and verification procedures.
- To understand the maintenance of software

UNIT-I:

Software and Software Engineering: The Nature of Software, The Unique Nature of WebApps, Software Engineering, Software Process, Software Engineering Practice, Software Myths.

Process Models: A Generic Process Model, Process Assessment and Improvement, Prescriptive Process Models, Specialized Process Models, The Unified Process, Personal and Team Process Models, Process Terminology, Product and Process.

UNIT-II:

Requirements Analysis And Specification: Requirements Gathering and Analysis, Software Requirement Specification (SRS), Formal System Specification.

Software Design: Overview of the Design Process, How to Characterise of a Design?, Cohesion and Coupling, Layered Arrangement of Modules, Approaches to Software Design

UNIT – III:

Function-Oriented Software Design: Overview of SA/SD Methodology, Structured Analysis, Developing the DFD Model of a System, Structured Design, Detailed Design, Design Review, over view of Object Oriented design.

User Interface Design: Characteristics of Good User Interface, Basic Concepts, Types of User Interfaces, Fundamentals of Component-based GUI Development, A User Interface Design Methodology.

UNIT – IV:

Coding And Testing: Coding, Code Review, Software Documentation, Testing, Unit Testing, Black-Box Testing, White-Box Testing, Debugging, Program Analysis Tool, Integration Testing, Testing Object-Oriented Programs, System Testing, Some General Issues Associated with Testing

UNIT – V:

Software Reliability And Quality Management: Software Reliability, Statistical Testing, Software Quality, Software Quality Management System, ISO 9000, SEI Capability Maturity Model.

Computer Aided Software Engineering: Case and its Scope, Case Environment, Case Support in Software Life Cycle, Other Characteristics of Case Tools, Towards Second Generation CASE Tool, Architecture of a Case Environment

UNIT – VI

Software Maintenance: Software maintenance, Maintenance Process Models, Maintenance Cost, Software Configuration Management.

Software Reuse: what can be reused? Why almost No Reuse So Far? Basic Issues in Reuse Approach, Reuse at Organization Level.

OUTCOMES

- Define and develop a software project from requirement gathering to implementation.
- Obtain knowledge about principles and practices of software engineering.
- Focus on the fundamentals of modeling a software project.
- Obtain knowledge about estimation and maintenance of software systems

TEXT BOOKS:

1. Software engineering A practitioner's Approach, Roger S. Pressman, Seventh Edition McGrawHill International Edition.
2. Fundamentals of Software Engineering, Rajib Mall, Third Edition, PHI.
3. Software Engineering, Ian Sommerville, Ninth edition, Pearson education

REFERENCE BOOKS:

1. Software Engineering : A Primer, Waman S Jawadekar, Tata McGraw-Hill, 2008
2. Software Engineering, A Precise Approach, Pankaj Jalote, Wiley India, 2010.
3. Software Engineering, Principles and Practices, Deepak Jain, Oxford University Press.
4. Software Engineering1: Abstraction and modeling, Diner Bjorner, Springer International edition, 2006.

II Year - II Semester

L	T	P	C
4	0	0	3

JAVA PROGRAMMING

OBJECTIVES:

- Understanding the OOP's concepts, classes and objects, threads, files, applets, swings and act.
- This course introduces computer programming using the JAVA programming language with object-oriented programming principles.
- Emphasis is placed on event-driven programming methods, including creating and manipulating objects, classes, and using Java for network level programming and middleware development

UNIT-I:

Introduction to OOP, procedural programming language and object oriented language, principles of OOP, applications of OOP, history of java, java features, JVM, program structure.
Variables, primitive data types, identifiers, literals, operators, expressions, precedence rules and associativity, primitive type conversion and casting, flow of control.

UNIT-II:

Classes and objects, class declaration, creating objects, methods, constructors and constructor overloading, garbage collector, importance of static keyword and examples, this keyword, arrays, command line arguments, nested classes.

UNIT-III:

Inheritance, types of inheritance, super keyword, final keyword, overriding and abstract class.
Interfaces, creating the packages, using packages, importance of CLASSPATH and java.lang package. Exception handling, importance of try, catch, throw, throws and finally block, user-defined exceptions, Assertions.

UNIT-IV:

Multithreading: introduction, thread life cycle, creation of threads, thread priorities, thread synchronization, communication between threads. Reading data from files and writing data to files, random access file,

UNIT-V:

Applet class, Applet structure, Applet life cycle, sample Applet programs. Event handling: event delegation model, sources of event, Event Listeners, adapter classes, inner classes.

UNIT-VI:
AWT: introduction, components and containers, Button, Label, Checkbox, Radio Buttons, List Boxes, Choice Boxes, Container class, Layouts, Menu and Scrollbar.

OUTCOMES:
- Understand Java programming concepts and utilize Java Graphical User Interface in Program writing.
- Write, compile, execute and troubleshoot Java programming for networking concepts.
- Build Java Application for distributed environment.
- Design and Develop multi-tier applications.
- Identify and Analyze Enterprise applications.

TEXT BOOKS:
1. The complete Reference Java, 8th edition, Herbert Schildt, TMH.
2. Programming in JAVA, Sachin Malhotra, SaurabhChoudary, Oxford.
3. Introduction to java programming, 7th edition by Y Daniel Liang, Pearson.

REFERENCE BOOKS:
1. Swing: Introduction, JFrame, JApplet, JPanel, Componets in Swings, Layout Managers in
2. Swings, JList and JScrollPane, Split Pane, JTabbedPane, JTree, JTable, Dialog Box.

L	T	P	C
4	0	0	3

II Year - II Semester

ADVANCED DATA STRUCTURES

OBJECTIVES:
- Describe and implement a variety of advanced data structures (hash tables, priority queues, balanced search trees, graphs).
- Analyze the space and time complexity of the algorithms studied in the course.
- Identify different solutions for a given problem; analyze advantages and disadvantages to different solutions.
- Demonstrate an understanding of external memory and external search and sorting algorithms.
- Demonstrate an understanding of simple Entity-Relationship models for databases.

UNIT-I: SORTING
External Sorting, Introduction, K-way Merging - Buffer Handling for parallel Operation- Run Generation- Optimal Merging of Runs.

UNIT-II: HASHING
Introduction-Static Hashing- Hash Table- Hash Functions- Secure Hash Function- Overflow Handling- Theoretical Evaluation of Overflow Techniques, Dynamic Hashing- Motivation for Dynamic Hashing -Dynamic Hashing Using Directories- Directory less Dynamic, Hashing,

UNIT-III:PRIORITY QUEUES (HEAPS)
Model, Simple Implementation, Binary Heap-Structure Property-Heap-Order Property-Basic Heap Operations- Other Heap Operation, Applications of Priority Queues- The Selection Problem Event Simulation Problem, Binomial Queues- Binomial Queue Structure – Binomial Queue Operation- Implementation of Binomial Queues

UNIT-IV: EFFICIENT BINARY SEARCH TREES
Optimal Binary Search Trees, AVL Trees, Red-Black Trees, Definition- Representation of a Red- Black Tree- Searching a Red-Black Tree- Inserting into a Red Black Tree- Deletion from a Red-Black Tree- Joining Red-Black Trees, Splitting a Red-Black tree.

UNIT-V: MULTIWAY SEARCH TREES
M-Way Search Trees, Definition and Properties- Searching an M-Way Search Tree, B-Trees, Definition and Properties- Number of Elements in a B-tree- Insertion into B-Tree- Deletion from a B-Tree- B+-Tree Definition- Searching a B+-Tree- Insertion into B+-tree- Deletion from a B+-Tree.

UNIT-VI: DIGITAL SEARCH STRUCTURES

Digital Search Trees, Definition- Search, Insert and Delete- Binary tries and Patricia, Binary Tries, Compressed Binary Tries- Patricia, Multiway Tries- Definitions- Searching a Trie- Sampling Strategies- Insertion into a Trie- Deletion from a Trie- Keys with Different Length- Height of a Trie- Space Required and Alternative Node Structure- Prefix Search and Applications- Compressed Tries- Compressed Tries With Skip Fields- Compressed Tries With Labeled Edges- Space Required by a Compressed Tries, Tries and Internet Packet Forwarding ,- IP Routing- 1-Bit Tries- Fixed-Stride Tries-Variable-Stride Tries.

OUTCOMES:
- Be able to understand and apply amortised analysis on data structures, including binary search trees, mergable heaps, and disjoint sets.
- Understand the implementation and complexity analysis of fundamental algorithms such as RSA, primality testing, max flow, discrete Fourier transform.
- Have an idea of applications of algorithms in a variety of areas, including linear programming and duality, string matching, game-theory

TEXT BOOKS:
1. Data Structures, a Pseudocode Approach, Richard F Gilberg, Behrouz A Forouzan, Cengage.
2. Fundamentals of DATA STRUCTURES in C: 2nd ed, , Horowitz , Sahani, Anderson-freed, Universities Press
3. Data structures and Algorithm Analysis in C, 2nd edition, Mark Allen Weiss, Pearson

REFERENCE BOOKS:
1. Web : http://lcm.csa.iisc.ernet.in/dsa/dsa.html
2. http://utubersity.com/?page_id=878
3. http://freevideolectures.com/Course/2519/C-Programming-and-Data-Structures
4. http://freevideolectures.com/Course/2279/Data-Structures-And-Algorithms
5. File Structures :An Object oriented approach with C++, 3rd ed, Michel J Folk, Greg Riccardi, Bill Zoellick
6. C and Data Structures: A Snap Shot oriented Treatise with Live examples from Science and Engineering, NB Venkateswarlu & EV Prasad, S Chand, 2010.

II Year - II Semester

L	T	P	C
4	0	0	3

COMPUTER ORGANIZATION

OBJECTIVES:

- Understand the architecture of a modern computer with its various processing units. Also the Performance measurement of the computer system.
- In addition to this the memory management system of computer.

UNIT -I:
Basic Structure Of Computers: Functional unit, Basic Operational concepts, Bus structures, System Software, Performance, The history of computer development.

UNIT -II:
Machine Instruction and Programs:
Instruction and Instruction Sequencing: Register Transfer Notation, Assembly Language Notation, Basic Instruction Types,
Addressing Modes, Basic Input/output Operations, The role of Stacks and Queues in computer programming equation. Component of Instructions: Logic Instructions, shift and Rotate Instructions

UNIT -III:
Type of Instructions: Arithmetic and Logic Instructions, Branch Instructions, Addressing Modes, Input/output Operations

UNIT -IV:
INPUT/OUTPUT ORGANIZATION: Accessing I/O Devices, Interrupts: Interrupt Hardware, Enabling and Disabling Interrupts, Handling Multiple Devices, Direct Memory Access,
Buses: Synchronous Bus, Asynchronous Bus, Interface Circuits, Standard I/O Interface: Peripheral Component Interconnect (PCI) Bus, Universal Serial Bus (USB)

UNIT -V:
The MEMORY SYSTEMS: Basic memory circuits, Memory System Consideration, Read-Only Memory: ROM, PROM, EPROM, EEPROM, Flash Memory,
Cache Memories: Mapping Functions, INTERLEAVING
Secondary Storage: Magnetic Hard Disks, Optical Disks,

UNIT -VI:

Processing Unit: Fundamental Concepts: Register Transfers, Performing An Arithmetic Or Logic Operation, Fetching A Word From Memory,

Execution of Complete Instruction, Hardwired Control,

Micro programmed Control: Microinstructions, Micro program Sequencing, Wide Branch Addressing Microinstructions with next –Address Field

OUTCOMES:
- Students can understand the architecture of modern computer.
- They can analyze the Performance of a   computer using performance equation
- Understanding of different instruction types.
- Students can calculate the effective address of an operand by addressing modes
- They can understand how computer stores positive and negative numbers.
- Understanding of how a computer performs arithmetic operation of positive and negative numbers.

TEXT BOOKS:

1. Computer Organization, Carl Hamacher, Zvonks Vranesic, Safea Zaky, 5th Edition, McGraw Hill.
 2. Computer Architecture and Organization, John P. Hayes, 3rd Edition, McGraw Hill.

REFERENCE BOOKS:

1. Computer Organization and Architecture – William Stallings Sixth Edition, Pearson/PHI
2. Structured Computer Organization – Andrew S. Tanenbaum, 4th Edition PHI/Pearson
3. Fundamentals or Computer Organization and Design, - Sivaraama Dandamudi Springer Int. Edition.
4. "Computer Organization and Design: The Hardware/Software Interface" by David A. Patterson and John L. Hennessy.
 5. J .P. Hayes, "Computer Architecture and Organization", McGraw-Hill, 1998.

	L	T	P	C

II Year – II Semester   L   T   P   C   4   0   0   3

FORMAL LANGUAGE AND AUTOMATA THEORY

OBJECTIVE:
- Introduce the student to the concepts of Theory of computation in computer science
- The students should acquire insights into the relationship among formal languages, formal Grammars and automat.

UNIT – I: Finite Automata

Why Study Automata Theory? The Central Concepts of Automata Theory, Automation, Finite Automation, Transition Systems, Acceptance of a String by a Finite Automation, DFA, Design of DFAs, NFA, Design of NFA, Equivalence of DFA and NFA, Conversion of NFA into DFA, Finite Automata with E-Transition, Minimization of Finite Automata, Mealy and Moore Machines, Applications and Limitation of Finite Automata.

UNIT – II: Regular Expressions

Regular Expressions, Regular Sets, Identity Rules, Equivalence of two Regular Expressions, Manipulations of Regular Expressions, Finite Automata, and Regular Expressions, Inter Conversion, Equivalence between Finite Automata and Regular Expressions, Pumping Lemma, Closers Properties, Applications of Regular Expressions, Finite Automata and Regular Grammars, Regular Expressions and Regular Grammars.

UNIT – III: Context Free Grammars

Formal Languages, Grammars, Classification of Grammars, Chomsky Hierarchy Theorem, Context Free Grammar, Leftmost and Rightmost Derivations, Parse Trees, Ambiguous Grammars, Simplification of Context Free Grammars-Elimination of Useless Symbols, E-Productions and Unit Productions, Normal Forms for Context Free Grammars-Chomsky Normal Form and Greibach Normal Form, Pumping Lemma, Closure Properties, Applications of Context Free Grammars.

UNIT – IV: Pushdown Automata

Pushdown Automata, Definition, Model, Graphical Notation, Instantaneous Description Language Acceptance of pushdown Automata, Design of Pushdown Automata, Deterministic and Non – Deterministic Pushdown Automata, Equivalence of Pushdown Automata and Context Free Grammars Conversion, Two Stack Pushdown Automata, Application of Pushdown Automata.

UNIT – V: Turning Machine

Turing Machine, Definition, Model, Representation of Turing Machines-Instantaneous Descriptions, Transition Tables and Transition Diagrams, Language of a Turing Machine, Design of Turing Machines, Techniques for Turing Machine Construction, Types of Turing Machines, Church's Thesis, Universal Turing Machine, Restricted Turing Machine.

UNIT – VI: Computability

Decidable and Un-decidable Problems, Halting Problem of Turing Machines, Post's Correspondence Problem, Modified Post's Correspondence Problem, Classes of P and NP, NP-Hard and NP-Complete Problems.

OUTCOMES:

- Classify machines by their power to recognize languages,
- Employ finite state machines to solve problems in computing,
- Explain deterministic and non-deterministic machines,
- Comprehend the hierarchy of problems arising in the computer science

TEXT BOOKS:

1. Introduction to Automata Theory, Languages and Computation, J.E.Hopcroft, R.Motwani and J.D.Ullman, 3rd Edition, Pearson, 2008.
2. Theory of Computer Science-Automata, Languages and Computation, K.L.P.Mishra and N.Chandrasekharan, 3rd Edition, PHI, 2007.

REFERENCE BOOKS:

1. Formal Language and Automata Theory, K.V.N.Sunitha and N.Kalyani, Pearson, 2015.
2. Introduction to Automata Theory, Formal Languages and Computation, Shyamalendu Kandar, Pearson, 2013.
3. Theory of Computation, V.Kulkarni, Oxford University Press, 2013.
    4. Theory of Automata, Languages and Computation, Rajendra Kumar, McGraw Hill, 2014.

L	T	P	C
4	0	0	3

PRINCIPLES OF PROGRAMMING LANGUAGES

OBJECTIVES:
- To understand and describe syntax and semantics of programming languages
- To understand data, data types, and basic statements
- To understand call-return architecture and ways of implementing them
- To understand object-orientation, concurrency, and event handling in programming languages
- To develop programs in non-procedural programming paradigms

UNIT- I:
Syntax and semantics: Evolution of programming languages, describing syntax, context, free grammars, attribute grammars, describing semantics, lexical analysis, parsing, recursive - decent bottom - up parsing

UNIT- II:
Data, data types, and basic statements: Names, variables, binding, type checking, scope, scope rules, lifetime and garbage collection, primitive data types, strings, array types, associative arrays, record types, union types, pointers and references, Arithmetic expressions, overloaded operators, type conversions, relational and boolean expressions , assignment statements , mixed mode assignments, control structures – selection, iterations, branching, guarded Statements

UNIT- III:
Subprograms and implementations: Subprograms, design issues, local referencing, parameter passing, overloaded methods, generic methods, design issues for functions, semantics of call and return, implementing simple subprograms, stack and dynamic local variables, nested subprograms, blocks, dynamic scoping

UNIT- IV:
Object- orientation, concurrency, and event handling: Object – orientation, design issues for OOP languages, implementation of object, oriented constructs, concurrency, semaphores, Monitors, message passing, threads, statement level concurrency, exception handling, event handling

UNIT -V:
Functional programming languages: Introduction to lambda calculus, fundamentals of functional programming languages, Programming with Scheme, –
Programming with ML,

UNIT -VI:
Logic programming languages: Introduction to logic and logic programming, –
Programming with Prolog, multi - paradigm languages

OUTCOMES:
- Describe syntax and semantics of programming languages
- Explain data, data types, and basic statements of programming languages
- Design and implement subprogram constructs, Apply object – oriented, concurrency, and event handling programming constructs
- Develop programs in Scheme, ML, and Prolog
- Understand and adopt new programming languages

TEXT BOOKS:
1. Robert W. Sebesta, "Concepts of Programming Languages", Tenth Edition, Addison Wesley, 2012.
2. Programming Langugaes, Principles & Paradigms, 2ed, Allen B Tucker, Robert E Noonan, TMH

REFERENCE BOOKS:
1. R. Kent Dybvig, "The Scheme programming language", Fourth Edition, MIT Press, 2009.
2. Jeffrey D. Ullman, "Elements of ML programming", Second Edition, Prentice Hall, 1998.
3. Richard A. O'Keefe, "The craft of Prolog", MIT Press, 2009.
4. W. F. Clocksin and C. S. Mellish, "Programming in Prolog: Using the ISO Standard", Fifth Edition, Springer, 2003

II Year – II Semester

L	T	P	C
0	0	3	2

ADVANCED DATA STRUCTURES LAB

OBJECTIVES:
- To understand heap and various tree structures like AVL, Red-black, B and Segment trees
- To understand the problems such as line segment intersection, convex shell and Voronoi diagram

Programming:

1. To perform various operations i.e., insertions and deletions on AVL trees.

2. To implement operations on binary heap.

      i) Vertex insertion

      ii) Vertex deletion

      iii) Finding vertex

      iv) Edge addition and deletion

3. To implement Prim's algorithm to generate a min-cost spanning tree.

4. To implement Krushkal's algorithm to generate a min-cost spanning tree.

5. To implement Dijkstra's algorithm to find shortest path in the graph.

6. To implementation of Static Hashing (Use Linear probing for collision resolution)

7. To implement of Huffmann coding.

8. To implement of B-tree.

OUTCOMES:
- Implement heap and various tree structure like AVL, Red-black, B and Segment trees
- Solve the problems such as line segment intersection, convex shell and Voronoi diagram

L	T	P	C
0	0	3	2

JAVA PROGRAMMING LAB

Exercise - 1 (Basics)

a). Write a JAVA program to display default value of all primitive data type of JAVA

b). Write a java program that display the roots of a quadratic equation $ax^2+bx=0$. Calculate the discriminate D and basing on value of D, describe the nature of root.

c). Five Bikers Compete in a race such that they drive at a constant speed which may or may not be the same as the other. To qualify the race, the speed of a racer must be more than the average speed of all 5 racers. Take as input the speed of each racer and print back the speed of qualifying racers.

d) Write a case study on **public static void main**(**250 words**)

Exercise - 2 (Operations, Expressions, Control-flow, Strings)

a). Write a JAVA program to search for an element in a given list of elements using binary search mechanism.

b). Write a JAVA program to sort for an element in a given list of elements using bubble sort

(c). Write a JAVA program to sort for an element in a given list of elements using merge sort.

(d) Write a JAVA program using StringBuffer to delete, remove character.

Exercise - 3 (Class, Objects)

a). Write a JAVA program to implement class mechanism. – Create a class, methods and invoke them inside main method.

b). Write a JAVA program to implement constructor.

Exercise - 4 (Methods)

a). Write a JAVA program to implement constructor overloading.

b). Write a JAVA program implement method overloading.

Exercise - 5 (Inheritance)

a). Write a JAVA program to implement Single Inheritance

b). Write a JAVA program to implement multi level Inheritance

c). Write a java program for abstract class to find areas of different shapes

Exercise - 6 (Inheritance - Continued)

a). Write a JAVA program give example for "super" keyword.

b). Write a JAVA program to implement Interface. What kind of Inheritance can be achieved?

Exercise - 7 (Exception)

a).Write a JAVA program that describes exception handling mechanism

b).Write a JAVA program Illustrating Multiple catch clauses

Exercise – 8 (Runtime Polymorphism)

a). Write a JAVA program that implements Runtime polymorphism

b). Write a Case study on run time polymorphism, inheritance that implements in above problem

Exercise – 9 (User defined Exception)

a). Write a JAVA program for creation of Illustrating throw

b). Write a JAVA program for creation of Illustrating finally

c). Write a JAVA program for creation of Java Built-in Exceptions

d).Write a JAVA program for creation of User Defined Exception

Exercise – 10 (Threads)

a). Write a JAVA program that creates threads by extending Thread class .First thread display "Good Morning "every 1 sec, the second thread displays "Hello "every 2 seconds and the third display "Welcome" every 3 seconds ,(Repeat the same by implementing Runnable)

b). Write a program illustrating **isAlive** and **join** ()

c). Write a Program illustrating Daemon Threads.

Exercise - 11 (Threads continuity)

a).Write a JAVA program Producer Consumer Problem

b).Write a case study on thread Synchronization after solving the above producer consumer problem

Exercise – 12 (Packages)

a). Write a JAVA program illustrate class path

b). Write a case study on including in class path in your os environment of your package.

c). Write a JAVA program that import and use the defined your package in the previous Problem

Exercise - 13 (Applet)

a).Write a JAVA program to paint like paint brush in applet.

b) Write a JAVA program to display analog clock using Applet.

c). Write a JAVA program to create different shapes and fill colors using Applet.

Exercise - 14 (Event Handling)

a).Write a JAVA program that display the x and y position of the cursor movement using

Mouse.

b).Write a JAVA program that identifies key-up key-down event user entering text in a
   Applet.

Exercise - 15 (Swings)
a).Write a JAVA programto build a Calculator in Swings
b). Write a JAVA program to display the digital watch in swing tutorial.

Exercise – 16 (Swings - Continued)
a). Write a JAVA program that to create a single ball bouncing inside a JPanel.
b). Write a JAVA program JTree as displaying a real tree upside down

	L	T	P	C
III Year – I Semester

L	T	P	C
4	0	0	3
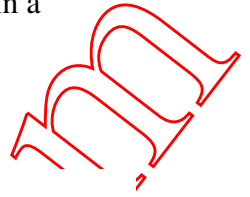
COMPILER DESIGN

OBJECTIVES:
- Understand the basic concept of compiler design, and its different phases which will be helpful to construct new tools like LEX, YACC, etc.

UNIT – I
Introduction Language Processing, Structure of a compiler the evaluation of Programming language, The Science of building a Compiler application of Compiler Technology. Programming Language Basics.
Lexical Analysis-: The role of lexical analysis buffing, specification of tokens. Recognitions of tokens the lexical analyzer generator lexical

UNIT –II
Syntax Analysis -: The Role of a parser, Context free Grammars Writing A grammar, top down passing bottom up parsing Introduction to Lr Parser.

UNIT –III
More Powerful LR parser (LR1, LALR) Using Armigers Grammars Equal Recovery in Lr parser Syntax Directed Transactions Definition, Evolution order of SDTS Application of SDTS. Syntax Directed Translation Schemes.

UNIT – IV
Intermediated Code: Generation Variants of Syntax trees 3 Address code, Types and Deceleration, Translation of Expressions, Type Checking. Canted Flow Back patching?

UNIT – V
Runtime Environments, Stack allocation of space, access to Non Local date on the stack Heap Management code generation – Issues in design of code generation the target Language Address in the target code Basic blocks and Flow graphs. A Simple Code generation.

UNIT –VI
Machine Independent Optimization. The principle sources of Optimization peep hole Optimization, Introduction to Date flow Analysis.

OUTCOMES:

- Acquire knowledge in different phases and passes of Compiler, and specifying different types of tokens by lexical analyzer, and also able to use the Compiler tools like LEX, YACC, etc.
- Parser and its types i.e. Top-down and Bottom-up parsers.
- Construction of LL, SLR, CLR and LALR parse table.
- Syntax directed translation, synthesized and inherited attributes.
- Techniques for code optimization.

TEXT BOOKS:

1. Compilers, Principles Techniques and Tools.Alfred V Aho, Monical S. Lam, Ravi Sethi Jeffery D. Ullman,2nd edition,pearson,2007

2. Compiler Design K.Muneeswaran, OXFORD

3. Principles of compiler design,2nd edition,Nandhini Prasad,Elsebier.

REFERENCE BOOKS:

1. Compiler Construction, Principles and practice, Kenneth C Louden, CENGAGE

2. Implementations of Compiler, A New approach to Compilers including the algebraic methods, Yunlinsu ,SPRINGER

L	T	P	C
4	0	0	3

UNIX PROGRAMMING

OBJECTIVES:

- Written technical communication and effective use of concepts and terminology.
- Facility with UNIX command syntax and semantics.
- Ability to read and understand specifications, scripts and programs.
- 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.

UNIT-I

Introduction to unix-Brief History-What is Unix-Unix Components-Using Unix-Commands in Unix-Some Basic Commands-Command Substitution-Giving Multiple Commands.

UNIT-II

The File system –The Basics of Files-What's in a File-Directories and File Names-Permissions-I Nodes-The Directory Hierarchy, File Attributes and Permissions-The File Command knowing the File Type-The Chmod Command Changing File Permissions-The Chown Command Changing the Owner of a File-The Chgrp Command Changing the Group of a File.

UNIT-III

Using the Shell-Command Line Structure-Met characters-Creating New Commands-Command Arguments and Parameters-Program Output as Arguments-Shell Variables- -More on I/O Redirection-Looping in Shell Programs.

UNIT-IV

Filters-The Grep Family-Other Filters-The Stream Editor Sed-The AWK Pattern Scanning and processing Language-Good Files and Good Filters.

UNIT-V

Shell Programming-Shell Variables-The Export Command-The Profile File a Script Run During Starting-The First Shell Script-The read Command-Positional parameters-The $? Variable knowing the exit Status-More about the Set Command-The Exit Command-Branching Control Structures-Loop Control Structures-The Continue and Break Statement-The Expr Command: Performing Integer Arithmetic-Real Arithmetic in Shell Programs-The here Document(<<)-The Sleep Command-Debugging Scripts-The Script Command-The Eval Command-The Exec Command.

UNIT-VI

The Process-The Meaning-Parent and Child Processes-Types of Processes-More about Foreground and Background processes-Internal and External Commands-Process Creation-The Trap Command-The Stty Command-The Kill Command-Job Control.

OUTCOMES:
- Documentation will demonstrate good organization and readability.
- File processing projects will require data organization, problem solving and research.
- Scripts and programs will demonstrate simple effective user interfaces.
- Scripts and programs will demonstrate effective use of structured programming.
- Scripts and programs will be accompanied by printed output demonstrating completion of a test plan.
- Testing will demonstrate both black and glass box testing strategies.
- Project work will involve group participation.

TEXT BOOKS:
1. The Unix programming Environment by Brain W. Kernighan & Rob Pike, Pearson.
2. Introduction to Unix Shell Programming by M.G.Venkateshmurthy, Pearson

REFERENCE BOOKS:
1. Unix and shell programmingby B.M. Harwani, OXFORD university press.

III Year – I Semester

L	T	P	C
4	0	0	3

OBJECT ORIENTED ANALYSIS & DESIGN USING UML

OBJECTIVE:
- To understand how to solve complex problems
- Analyze and design solutions to problems using object oriented approach
- Study the notations of Unified Modeling Language

UNIT-I:
Introduction: The Structure of Complex systems, The Inherent Complexity of Software, Attributes of Complex System, Organized and Disorganized Complexity, Bringing Order to Chaos, Designing Complex Systems, Evolution of Object Model, Foundation of Object Model, Elements of Object Model, Applying the Object Model.

UNIT-II:
Classes and Objects: Nature of object, Relationships among objects, Nature of a Class, Relationship among Classes, Interplay of Classes and Objects, Identifying Classes and Objects, Importance of Proper Classification, Identifying Classes and Objects, Key abstractions and Mechanisms.

UNIT-III:
Introduction to UML: Why we model, Conceptual model of UML, Architecture, Classes, Relationships, Common Mechanisms, Class diagrams, Object diagrams.

UNIT-IV:
Basic Behavioral Modeling: Interactions, Interaction diagrams, Use cases, Use case Diagrams, Activity Diagrams.

UNIT-V:
Advanced Behavioral Modeling: Events and signals, state machines, processes and Threads, time and space, state chart diagrams.

UNIT-VI:
Architectural Modeling: Component, Deployment, Component diagrams and Deployment diagrams.
Case Study: The Unified Library application.

OUTCOME:
- Ability to find solutions to the complex problems using object oriented approach
- Represent classes, responsibilities and states using UML notation
- Identify classes and responsibilities of the problem domain

TEXT BOOKS:

1. "Object- Oriented Analysis And Design with Applications", Grady BOOCH, Robert A. Maksimchuk, Michael W. ENGLE, Bobbi J. Young, Jim Conallen, Kellia Houston, 3rd edition, 2013, PEARSON.
2. "The Unified Modeling Language User Guide", Grady Booch, James Rumbaugh, Ivar Jacobson, 12th Impression, 2012, PEARSON.

REFERENCE BOOKS:

1. "Object-oriented analysis and design using UML", Mahesh P. Matha, PHI
2. "Head first object-oriented analysis and design", Brett D. McLaughlin, Gary Pollice, Dave West, O"Reilly
3. "Object-oriented analysis and design with the Unified process", John W. Satzinger, Robert B. Jackson, Stephen D. Burd, Cengage Learning
1. "The Unified modeling language Reference manual", James Rumbaugh, Ivar Jacobson, Grady Booch, Addison-Wesley

	L	T	P	C
III Year – I Semester   4   0   0   3

DATA BASE MANAGEMENT SYSTEMS

OBJECTIVES

- To learn the principles of systematically designing and using large scale Database Management Systems for various applications.

UNIT-I: An Overview of Database Management, Introduction- What is Database System- What is Database-Why Database- Data Independence- Relation Systems and Others- Summary, **Database system architecture, Introduction-** The Three Levels of Architecture-The External Level- the Conceptual Level- the Internal Level- Mapping- the Database Administrator-The Database Management Systems- Client/Server Architecture.

UNIT-II:

The E/R Models, The Relational Model, Relational Calculus, Introduction to Database Design, Database Design and Er Diagrams-Entities Attributes, and Entity Sets-Relationship and Relationship Sets-Conceptual Design With the Er Models, The Relational Model Integrity Constraints Over Relations- Key Constraints –Foreign Key Constraints-General Constraints, Relational Algebra and Calculus, Relational Algebra- Selection and Projection- Set Operation, Renaming – Joins- Division- More Examples of Queries, Relational Calculus, Tuple Relational Calculus- Domain Relational Calculus.

UNIT-III:

Queries, Constraints, Triggers: The Form of Basic SQL Query, Union, Intersect, and Except, Nested Queries, Aggregate Operators, Null Values, Complex Integrity Constraints in SQL, Triggers and Active Database.

UNIT-IV:

Schema Refinement (Normalization) : Purpose of Normalization or schema refinement, concept of functional dependency, normal forms based on functional dependency(1NF, 2NF and 3 NF), concept of surrogate key, Boyce-codd normal form(BCNF), Lossless join and dependency preserving decomposition, Fourth normal form(4NF).

UNIT-V:
Transaction Management and Concurrency Control:

Transaction, properties of transactions, transaction log, and transaction management with SQL using commit rollback and save point.

Concurrency control for lost updates, uncommitted data, inconsistent retrievals and the Scheduler. Concurrency control with locking methods : lock granularity, lock types, two phase locking for ensuring serializability, deadlocks, Concurrency control with time stamp ordering : Wait/Die and Wound/Wait Schemes, Database Recovery management : Transaction recovery.

UNIT-VI:

Overview of Storages and Indexing, Data on External Storage- File Organization and Indexing – Clustered Indexing – Primary and Secondary Indexes, Index Data Structures, Hash-Based Indexing – Tree-Based Indexing, Comparison of File Organization

OUTCOMES
- Describe a relational database and object-oriented database.
- Create, maintain and manipulate a relational database using SQL
- Describe ER model and normalization for database design.
- Examine issues in data storage and query processing and can formulate appropriate solutions.
- Understand the role and issues in management of data such as efficiency, privacy, security, ethical responsibility, and strategic advantage.
- Design and build database system for a given real world problem

TEXT BOOKS:
1. Introduction to Database Systems, CJ Date, Pearson
2. Data base Management Systems, Raghurama Krishnan, Johannes Gehrke, TATA McGraw Hill 3rd Edition
3. Database Systems - The Complete Book, H G Molina, J D Ullman, J Widom Pearson

REFERENCES BOOKS:
1. Data base Systems design, Implementation, and Management, Peter Rob & Carlos Coronel 7th Edition.
2. Fundamentals of Database Systems, Elmasri Navrate Pearson Education
   3. Introduction to Database Systems, C.J.Date Pearson Education

L	T	P	C
4	0	0	3

III Year – I Semester

OPERATING SYSTEMS

OBJECTIVES:

- Study the basic concepts and functions of operating systems.
- Understand the structure and functions of OS.
- Learn about Processes, Threads and Scheduling algorithms.
- Understand the principles of concurrency and Deadlocks.
- Learn various memory management schemes.
- Study I/O management and File systems.
- Learn the basics of Linux system and perform administrative tasks on Linux Servers.

UNIT I
Introduction to Operating System Concept: Types of operating systems, operating systems concepts, operating systems services, Introduction to System call, System call types.

UNIT-II:
Process Management – Process concept, The process, Process State Diagram , Process control block, Process Scheduling- Scheduling Queues, Schedulers, Operations on Processes, Interprocess Communication, Threading Issues, Scheduling-Basic Concepts, Scheduling Criteria, Scheduling Algorithms.

UNIT-III:
Memory Management: Swapping, Contiguous Memory Allocation, Paging, structure of the Page Table, Segmentation
Virtual Memory Management:
Virtual Memory, Demand Paging, Page-Replacement Algorithms, Thrashing

UNIT-IV:
Concurrency: ProcessSynchronization, The Critical- Section Problem, Synchronization Hardware, Semaphores, Classic Problems of Synchronization, Monitors, Synchronization examples
Principles of deadlock – System Model, Deadlock Characterization, Deadlock Prevention, Detection and Avoidance, Recovery form Deadlock

UNIT-V:

File system Interface- the concept of a file, Access Methods, Directory structure, File system mounting, file sharing, protection.

File System implementation- File system structure, allocation methods, free-space management

Mass-storage structure overview of Mass-storage structure, Disk scheduling, Device drivers,

UNIT VI:

Linux System: Components of LINUX, Interprocess Communication, Synchronisation, Interrupt, Exception and System Call.

Android Software Platform: Android Architecture, Operating System Services, Android Runtime Application Development, Application Structure, Application Process management

OUTCOMES:

- Design various Scheduling algorithms.

- Apply the principles of concurrency.

- Design deadlock, prevention and avoidance algorithms.

- Compare and contrast various memory management schemes.

- Design and Implement a prototype file systems.

- Perform administrative tasks on Linux Servers

- Introduction to Android Operating System Internals

TEXT BOOK:

1. Operating System Concepts, Abraham Silberschatz, Peter Baer Galvin and Greg Gagne 9th Edition, John Wiley and Sons Inc., 2012.
2. Operating Systems – Internals and Design Principles, William Stallings, 7th Edition, Prentice Hall, 2011.
3. Operating Systems-S Halder, Alex A Aravind Pearson Education Second Edition 2016 .

REFERENCES:

1. Modern Operating Systems, Andrew S. Tanenbaum, Second Edition, Addison Wesley, 2001.
2. Operating Systems: A Design-Oriented Approach, Charles Crowley, Tata Mc Graw Hill Education", 1996.
3. Operating Systems: A Concept-Based Approach, D M Dhamdhere, Second Edition, Tata Mc Graw-Hill Education, 2007.

L	T	P	C
0	0	3	2

III Year – I Semester

UNIFIED MODELING LAB

OBJECTIVES:
- Construct UML diagrams for static view and dynamic view of the system.
- Generate creational patterns by applicable patterns for given context.
- Create refined model for given Scenario using structural patterns.
- Construct behavioral patterns for given applications.

Week 1:
Familiarization with Rational Rose or Umbrello

For each case study:

Week 2, 3 & 4:
For each case study:
- a) Identify and analyze events
- b) Identify Use cases
- c) Develop event table
- d) Identify & analyze domain classes
- e) Represent use cases and a domain class diagram using Rational Rose
- f) Develop CRUD matrix to represent relationships between use cases and problem domain classes

Week 5 & 6:
- **For each case study:**
- a) Develop Use case diagrams
- b) Develop elaborate Use case descriptions & scenarios
- c) Develop prototypes (without functionality)
- d) Develop system sequence diagrams

Week 7, 8, 9 & 10:
For each case study:
- a) Develop high-level sequence diagrams for each use case
- b) Identify MVC classes / objects for each use case
- c) Develop Detailed Sequence Diagrams / Communication diagrams for each use case showing interactions among all the three-layer objects
- d) Develop detailed design class model (use GRASP patterns for responsibility assignment)
- e) Develop three-layer package diagrams for each case study

Week 11 & 12:
- **For each case study:**
- a) Develop Use case Packages
- b) Develop component diagrams
- c) Identify relationships between use cases and represent them
- d) Refine domain class model by showing all the associations among classes

- **Week 13 onwards:**
- **For each case study:**

- a) Develop sample diagrams for other UML diagrams - state chart diagrams, activity diagrams and deployment diagrams

  OUTCOMES:
- Understand the Case studies and design the Model.
- Understand how design patterns solve design problems.
- Develop design solutions using creational patterns.

Construct design solutions by using structural and behavioral patterns

L	T	P	C
0	0	3	2

OPERATING SYSEMS AND LINUX PROGRAMMING LAB

OBJECTIVES:

- To understand the design aspects of operating system.
- To study the process management concepts & Techniques.
- To study the storage management concepts.
- To familiarize students with theLinux environment
- To learn the fundamentals of shell scripting/programming
- To conceptualize Data Mining and the need for pre-processing.
- To learn the algorithms used for various types of Data Mining Problem

OPERATING SYSTEMS

1. Simulate the following CPU scheduling algorithms

   a) Round Robin b) SJF c) FCFS d) Priority

2. Multiprogramming-Memory management- Implementation of fork (), wait (), exec() and exit (), System calls

3. Simulate the following

   a) Multiprogramming with a fixed number of tasks (MFT)

   b) Multiprogramming with a variable number of tasks (MVT)

4. Simulate Bankers Algorithm for Dead Lock Avoidance

5. Simulate Bankers Algorithm for Dead Lock Prevention.

6. Simulate the following page replacement algorithms.

   a) FIFO b) LRU c) LFU

7. Simulate the following File allocation strategies

   a) Sequenced b) Indexed c) Linked

LINUX PROGRAMMING

1. a) Study 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.

   b) Study of vi editor.

   c) Study of Bash shell, Bourne shell and C shell in Unix/Linux operating system.

   d) Study of Unix/Linux file system (tree structure).

   e) Study of .bashrc, /etc/bashrc and Environment variables.

2. Write a C program that makes a copy of a file using standard I/O, and system calls

3. Write a C program to emulate the UNIX ls –l command.

4. Write a C program that illustrates how to execute two commands concurrently

   with a command pipe.

               Ex: - ls –l | sort

5. Write a C program that illustrates two processes communicating using sharedmemory

6. Write a C program to simulate producer and consumer problem usingsemaphores

7. Write C program to create a thread using pthreads library and let it run its function.

8. Write a C program to illustrate concurrent execution of threads using pthreads library.

OUTCOMES:

- To use Unix utilities and perform basic shell control of the utilities
- To use the Unix file system and file access control.
- To use of an operating system to develop software
- Students will be able to use Linux environment efficiently
- Solve problems using bash for shell scripting
- Will be able to implement algorithms to solve data mining problems using weka tool

L	T	P	C
0	0	3	2

DATA BASE MANAGEMENT SYSTEM LAB

OBJECTIVES:

- To provide a sound introduction to the discipline of database management as a subject in its own right, rather than as a compendium of techniques and product-specific tools.

- To familiarize the participant with the nuances of database environments towards an information-oriented data-processing oriented framework

- To give a good formal foundation on the relational model of data

- To present SQL and procedural interfaces to SQL comprehensively

- To give an introduction to systematic database design approaches covering conceptual design, logical design and an overview of physical design

List of Experiments:

SQL

1. Queries to facilitate acquaintance of Built-In Functions, String Functions, Numeric Functions, Date Functions and Conversion Functions.

2. Queries using operators in SQL

3. Queries to Retrieve and Change Data: Select, Insert, Delete, and Update

4. Queries using Group By, Order By, and Having Clauses

5. Queries on Controlling Data: Commit, Rollback, and Save point

6. Queries to Build Report in SQL *PLUS

7. Queries for Creating, Dropping, and Altering Tables, Views, and Constraints

8. Queries on Joins and Correlated Sub-Queries

9. Queries on Working with Index, Sequence, Synonym, Controlling Access, and Locking Rows for Update, Creating Password and Security features

PL/SQL

10. Write a PL/SQL Code using Basic Variable, Anchored Declarations, and Usage of

Assignment Operation

11. Write a PL/SQL Code Bind and Substitution Variables. Printing in PL/SQL

12. Write a PL/SQL block using SQL and Control Structures in PL/SQL

13. Write a PL/SQL Code using Cursors, Exceptions and Composite Data Types

14. Write a PL/SQL Code using Procedures, Functions, and Packages FORMS

15. Write a PL/SQL Code Creation of forms for any Information System such as Student Information System, Employee Information System etc. 18

16. Demonstration of database connectivity

OUTCOMES:
- Understand, appreciate and effectively explain the underlying concepts of database technologies
- Design and implement a database schema for a given problem-domain
- Normalize a database
- Populate and query a database using SQL DML/DDL commands.
- Declare and enforce integrity constraints on a database using a state-of-the-artRDBMS
- Programming PL/SQL including stored procedures, stored functions, cursors, packages.
- Design and build a GUI application using a 4GL

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

Text Books/Suggested Reading:
1. Oracle: The Complete Reference by Oracle Press
2. Nilesh Shah, "Database Systems Using Oracle", PHI, 2007.
3. Rick F Vander Lans, "Introduction to SQL", Fourth Edition, Pearson Education, 2007

	L	**T**	**P**	**C**
III Year – I Semester	**0**	**3**	**0**	**0**

PROFESSIONAL ETHICSAND HUMAN VALUES

Course Objectives:

*To give basic insights and inputs to the student to inculcate Human values to grow as a responsible human beings with proper personality.
*Professional Ethics instills the student to maintain ethical conduct and discharge their professional duties.

UNIT I: Human Values:
Morals, Values and Ethics – Integrity –Trustworthiness - Work Ethics – Service Learning – Civic Virtue – Respect for others – Living Peacefully – Caring – Sharing – Honesty –Courage – Value Time – Co-operation – Commitment – Empathy – Self-confidence – Spirituality-Character.

UNIT: II: Principles for Harmony:
Truthfulness – Customs and Traditions -Value Education – Human Dignity – Human Rights – Fundamental Duties - Aspirations and Harmony (I, We & Nature) – Gender Bias - Emotional Intelligence – Salovey – Mayer Model - Emotional Competencies – Conscientiousness.

UNIT III: Engineering Ethics and Social Experimentation:
History of Ethics - Need of Engineering Ethics - Senses of Engineering Ethics- Profession and Professionalism —Self Interest - Moral Autonomy – Utilitarianism – Virtue Theory - Uses of Ethical Theories - Deontology- Types of Inquiry –Kohlberg's Theory - Gilligan's Argument – Heinz's Dilemma - Comparison with Standard Experiments — Learning from the Past – Engineers as Managers –Consultants and Leaders – Balanced Outlook on Law - Role of Codes – Codes and Experimental Nature of Engineering.

UNIT IV: Engineers' Responsibilities towards Safety and Risk:

Concept of Safety - Safety and Risk – Types of Risks – Voluntary v/sInvoluntary Risk – Consequences - Risk Assessment – Accountability – Liability - Reversible Effects - Threshold Levels of Risk - Delayed v/sImmediate Risk - Safety and the Engineer – Designing for Safety – Risk-Benefit Analysis-Accidents.

UNIT V: Engineers' Duties and Rights:

Concept of Duty - Professional Duties – Collegiality - Techniques for Achieving Collegiality – Senses of Loyalty - Consensus and Controversy - Professional and Individual Rights – Confidential and Proprietary Information - Conflict of Interest-Ethical egoism - Collective Bargaining – Confidentiality - Gifts and Bribes - Problem solving-Occupational Crimes- Industrial Espionage- Price Fixing-Whistle Blowing.

UNIT VI: Global Issues:
Globalization and MNCs –Cross Culture Issues - Business Ethics – Media Ethics - Environmental Ethics – Endangering Lives - Bio Ethics - Computer Ethics - War Ethics – Research Ethics -Intellectual Property Rights.

- Related Cases Shall be dealt where ever necessary.

Outcome:
*It gives a comprehensive understanding of a variety issues that are encountered by every professional in discharging professional duties.
*It provides the student the sensitivity and global outlook in the contemporary world to fulfill the professional obligations effectively.

References:

1. Professional Ethics by R. Subramaniam – Oxford Publications, New Delhi.
2. Ethics in Engineering by Mike W. Martin and Roland Schinzinger - Tata McGraw-Hill – 2003.
3. Professional Ethics and Morals by Prof.A.R.Aryasri, DharanikotaSuyodhana - Maruthi Publications.
4. Engineering Ethics by Harris, Pritchard and Rabins, Cengage Learning, New Delhi.
5. Human Values & Professional Ethics by S. B. Gogate, Vikas Publishing House Pvt. Ltd., Noida.
6. Engineering Ethics & Human Values by M.Govindarajan, S.Natarajan and V.S.SenthilKumar-PHI Learning Pvt. Ltd – 2009.
7. Professional Ethics and Human Values by A. Alavudeen, R.Kalil Rahman and M. Jayakumaran – University Science Press.
8. Professional Ethics and Human Values by Prof.D.R.Kiran-Tata McGraw-Hill - 2013
9. Human Values And Professional Ethics by Jayshree Suresh and B. S. Raghavan, S.Chand Publications

	L	**T**	**P**	**C**

III Year – II Semester

L   T   P   C
4   2   0   3

COMPUTER NETWORKS

OBJECTIVES:

- Understand state-of-the-art in network protocols, architectures, and applications.
- Process of networking research
- Constraints and thought processes for networking research
- Problem Formulation—Approach—Analysis—

UNIT – I:
Introduction: Network Topologies WAN, LAN, MAN. Reference models- The OSI Reference Model- the TCP/IP Reference Model - A Comparison of the OSI and TCP/IP Reference Models

UNIT – II:
Physical Layer – Fourier Analysis – Bandwidth Limited Signals – The Maximum Data Rate of a Channel - Guided Transmission Media, Digital Modulation and Multiplexing: Frequency Division Multiplexing, Time Division Multiplexing, Code Division Multiplexing
Data Link Layer Design Issues, Error Detection and Correction, Elementary Data Link Protocols, Sliding Window Protocols

UNIT – III:
The Data Link Layer - Services Provided to the Network Layer – Framing – Error Control – Flow Control, Error Detection and Correction – Error-Correcting Codes – Error Detecting Codes, Elementary Data Link Protocols- A Utopian Simplex Protocol-A Simplex Stop and Wait Protocol for an Error free channel-A Simplex Stop and Wait Protocol for a Noisy Channel, Sliding Window Protocols-A One Bit Sliding Window Protocol-A Protocol Using Go-Back-N-A Protocol Using Selective Repeat

UNIT – IV:
The Medium Access Control Sublayer-The Channel Allocation Problem-Static Channel Allocation-Assumptions for Dynamic Channel Allocation, Multiple Access Protocols-Aloha-Carrier Sense Multiple Access Protocols-Collision-Free Protocols-Limited Contention Protocols-Wireless LAN Protocols, Ethernet-Classic Ethernet Physical Layer-Classic Ethernet MAC Sublayer Protocol-Ethernet Performance-Fast Ethernet Gigabit Ethernet-10-Gigabit Ethernet-Retrospective on Ethernet, Wireless Lans-The 802.11 Architecture and Protocol Stack-The 802.11 Physical Layer-The802.11 MAC Sublayer Protocol-The 805.11 Frame Structure-Services

UNIT – V:
Design Issues-The Network Layer Design Issues – Store and Forward Packet Switching-Services Provided to the Transport layer- Implementation of Connectionless Service-Implementation of Connection Oriented Service-Comparison of Virtual Circuit and Datagram Networks, Routing Algorithms-The Optimality principle-Shortest path Algorithm, Congestion Control Algorithms-Approaches to Congestion Control-Traffic Aware Routing-Admission Control-Traffic Throttling-Load Shedding.

UNIT – VI:
Transport Layer – The Internet Transport Protocols: Udp, the Internet Transport Protocols: Tcp
Application Layer –The Domain Name System: The DNS Name Space, Resource Records, Name Servers, Electronic Mail: Architecture and Services, The User Agent, Message Formats, Message Transfer, Final Delivery

OUTCOMES:
- Understand OSI and TCP/IP models
- Analyze MAC layer protocols and LAN technologies
- Design applications using internet protocols
- Understand routing and congestion control algorithms
- Understand how internet works

TEXT BOOKS:
1. Tanenbaum and David J Wetherall, Computer Networks, 5th Edition, Pearson Edu, 2010
2. Computer Networks: A Top Down Approach, Behrouz A. Forouzan, FirouzMosharraf, McGraw Hill Education

REFERENCE BOOKS:
1. Larry L. Peterson and Bruce S. Davie, "Computer Networks - A Systems Approach" (5th ed), Morgan Kaufmann/ Elsevier, 2011

III Year – II Semester

L	T	P	C
4	0	0	3

DATA WARE HOUSING AND DATA MINING

OBJECTIVES:
- Students will be enabled to understand and implement classical models and algorithms in data warehousing and data mining.
- They will learn how to analyze the data, identify the problems, and choose the relevant models and algorithms to apply.
- They will further be able to assess the strengths and weaknesses of various methods and algorithms and to analyze their behavior.

UNIT –I:
Introduction:Why Data Mining? What Is Data Mining?1.3 What Kinds of Data Can Be Mined?1.4 What Kinds of Patterns Can Be Mined?Which Technologies Are Used?Which Kinds of Applications Are Targeted?Major Issues in Data Mining.Data Objects and Attribute Types,Basic Statistical Descriptions of Data,Data Visualization, Measuring Data Similarity and Dissimilarity

UNIT –II:
Data Pre-processing: Data Preprocessing: An Overview,Data Cleaning,Data Integration,Data Reduction,Data Transformation and Data Discretization

UNIT –III:
Classification: Basic Concepts, General Approach to solving a classification problem, Decision Tree Induction: Working of Decision Tree, building a decision tree, methods for expressing an attribute test conditions, measures for selecting the best split, Algorithm for decision tree induction.

UNIT –IV:
Classification: Alterative Techniques, Bayes' Theorem, Naïve Bayesian Classification, Bayesian Belief Networks

UNIT –V
Association Analysis: Basic Concepts and Algorithms: Problem Defecation, Frequent Item Set generation, Rule generation, compact representation of frequent item sets, FP-Growth Algorithm. **(Tan &Vipin)**

UNIT –VI
Cluster Analysis: Basic Concepts and Algorithms:Overview: What Is Cluster Analysis? Different Types of Clustering, Different Types of Clusters; K-means: The Basic K-means Algorithm, K-means Additional Issues, Bisecting K-means, Strengths and Weaknesses; Agglomerative Hierarchical Clustering: Basic Agglomerative Hierarchical Clustering Algorithm DBSCAN: Traditional Density Center-Based Approach, DBSCAN Algorithm, Strengths and Weaknesses. **(Tan &Vipin)**

OUTCOMES:
- Understand stages in building a Data Warehouse
- Understand the need and importance of preprocessing techniques
- Understand the need and importance of Similarity and dissimilarity techniques
- Analyze and evaluate performance of algorithms for Association Rules.
- Analyze Classification and Clustering algorithms

TEXT BOOKS:
1. Introduction to Data Mining: Pang-Ning Tan & Michael Steinbach, Vipin Kumar, Pearson.
2. Data Mining concepts and Techniques, 3/e, Jiawei Han, Michel Kamber, Elsevier.

REFERENCE BOOKS:
1. Data Mining Techniques and Applications: An Introduction, Hongbo Du, Cengage Learning.
2. Data Mining : VikramPudi and P. Radha Krishna, Oxford.
3. Data Mining and Analysis - Fundamental Concepts and Algorithms: Mohammed J. Zaki, Wagner Meira, Jr, Oxford
4. Data Warehousing Data Mining & OLAP, Alex Berson, Stephen Smith, TMH.

III Year – II Semester

L	T	P	C
4	0	0	3

DESIGN AND ANALYSIS OF ALGORITHMS

OBJECTIVES:

Upon completion of this course, students will be able to do the following:

- Analyze the asymptotic performance of algorithms.
- Write rigorous correctness proofs for algorithms.
- Demonstrate a familiarity with major algorithms and data structures.
- Apply important algorithmic design paradigms and methods of analysis.
- Synthesize efficient algorithms in common engineering design situations

UNIT-I:

Introduction: What is an Algorithm, Algorithm Specification, Pseudocode Conventions Recursive Algorithm, Performance Analysis, Space Complexity, Time Complexity, Amortized Complexity, Amortized Complexity, Asymptotic Notation, Practical Complexities, Performance Measurement.

UNIT-II:

Dived and Conquer: General Method, Defective Chessboard, Binary Search, Finding the Maximum and Minimum, Merge Sort, Quick Sort, Performance Measurement, Randomized Sorting Algorithms.

UNIT-III:

The Greedy Method: The General Method, Knapsack Problem, Job Sequencing with Deadlines, Minimum-cost Spanning Trees, Prim's Algorithm, Kruskal's Algorithms, An Optimal Randomized Algorithm, Optimal Merge Patterns, Single Source Shortest Paths.

UNIT-IV:

Dynamic Programming: All - Pairs Shortest Paths, Single – Source Shortest paths General Weights, String Edition, 0/1 Knapsack, Reliability Design,

UNIT-V:

Backtracking: The General Method, The 8-Queens Problem, Sum of Subsets, Graph Coloring , Hamiltonian Cycles.

UNIT-VI:

Branch and Bound: The Method, Least cost (LC) Search, The 15-Puzzle: an Example, Control Abstraction for LC-Search, Bounding, FIFO Branch-and-Bound, LC Branch and Bound, 0/1 Knapsack Problem, LC Branch-and Bound Solution, FIFO Branch-and-Bound Solution, Traveling Salesperson.

OUTCOMES:

Students who complete the course will have demonstrated the ability to do the following:

- Argue the correctness of algorithms using inductive proofs and invariants.
- Analyze worst-case running times of algorithms using asymptotic analysis.
- Describe the divide-and-conquer paradigm and explain when an algorithmic design situation calls for it. Recite algorithms that employ this paradigm. Synthesize divide-and-conquer algorithms. Derive and solve recurrences describing the performance of divide-and-conquer algorithms.
- Describe the dynamic-programming paradigm and explain when an algorithmic design situation calls for it. Recite algorithms that employ this paradigm. Synthesize dynamic-programming algorithms, and analyze them.
- Describe the greedy paradigm and explain when an algorithmic design situation calls for it. Recite algorithms that employ this paradigm. Synthesize greedy algorithms, and analyze them.

TEXT BOOKS:

1. Fundamentals of computer algorithms E. Horowitz S. Sahni, University Press
2. Introduction to AlgorithmsThomas H. Cormen, PHI Learning

REFERENCE BOOKS

1. The Design and Analysis of Computer Algorithms, Alfred V. Aho, John E. Hopcroft, Jeffrey D. Ullman
2. Algorithm Design, Jon Kleinberg, Pearson.

III Year – II Semester

L	T	P	C
4	0	0	3

SOFTWARE TESTING METHODOLOGIES

OBJECTIVE:

Fundamentals for various testing methodologies.

- Describe the principles and procedures for designing test cases.
- Provide supports to debugging methods.
- Acts as the reference for software testing techniques and strategies.

UNIT-I:

Introduction: Purpose of Testing, Dichotomies, Model for Testing, Consequences of Bugs,Taxonomy of Bugs.

Flow graphs and Path testing: Basics Concepts of Path Testing, Predicates, Path Predicates andAchievable Paths, Path Sensitizing, Path Instrumentation, Application of Path Testing.

UNIT-II:

Transaction Flow Testing: Transaction Flows, Transaction Flow Testing Techniques.

Dataflow testing: Basics of Dataflow Testing, Strategies in Dataflow Testing, Application ofDataflow Testing.

UNIT-III:

Domain Testing: Domains and Paths, Nice & Ugly Domains, Domain testing, Domains andInterfaces Testing, Domain and Interface Testing, Domains and Testability.

Paths, Path products and Regular expressions: Path Products & Path Expression, Reduction Procedure, Applications, Regular Expressions & Flow Anomaly Detection.

UNIT-IV:

Syntax Testing: Why, What and How, A Grammar for formats, Test Case Generation, Implementation and Application and Testability Tips.

Logic Based Testing: Overview, Decision Tables, Path Expressions, KV Charts, and Specifications.

UNIT – V:

State, State Graphs and Transition Testing: State Graphs, Good & Bad State Graphs, State Testing, and Testability Tips.

Graph Matrices and Application:-Motivational overview, matrix of graph, relations, power of a matrix, node reduction algorithm.

UNIT -VI:
Software Testing Tools: Introduction to Testing, Automated Testing, Concepts of Test Automation, Introduction to list of tools like Win runner, Load Runner, Jmeter, About Win Runner ,Using Win runner, Mapping the GUI, Recording Test, Working with Test, Enhancing Test, Checkpoints, Test Script Language, Putting it all together, Running and Debugging Tests, Analyzing Results, Batch Tests, Rapid Test Script Wizard.

OUTCOME:
- Understand the basic testing procedures.

- Able to support in generating test cases and test suites.

- Able to test the applications manually by applying different testing methods and automation tools.
- Apply tools to resolve the problems in Real time environment.

TEXT BOOKS:
1. Software testing techniques – Boris Beizer, Dreamtech, second edition.

2. Software Testing- Yogesh Singh, Camebridge

REFERENCE BOOKS:

1. The Craft of software testing - Brian Marick, Pearson Education.

2. Software Testing, 3rd edition, P.C. Jorgensen, Aurbach Publications (Dist.by SPD).
3. Software Testing, N.Chauhan, Oxford University Press.

4. Introduction to Software Testing, P.Ammann&J.Offutt, Cambridge Univ.Press.

5. Effective methods of Software Testing, Perry, John Wiley, 2nd Edition, 1999.
6. Software Testing Concepts and Tools, P.NageswaraRao, dreamtech Press
7. Win Runner in simple steps by Hakeem Shittu, 2007Genixpress.
8. Foundations of Software Testing, D.Graham& Others, Cengage Learning.

III Year – II Semester

L	T	P	C
4	0	0	3

ARTIFICIAL INTELLIGENCE
(Open Elective)

OBJECTIVES:
- To have a basic proficiency in a traditional AI language including an ability to write simple to intermediate programs and an ability to understand code written in that language.
- To have an understanding of the basic issues of knowledge representation and blind and heuristic search, as well as an understanding of other topics such as minimax, resolution, etc. that play an important role in AI programs.
- To have a basic understanding of some of the more advanced topics of AI such as learning, natural language processing, agents and robotics, expert systems, and planning

UNIT-I:
Introduction to artificial intelligence: Introduction ,history, intelligent systems, foundations of AI, applications, tic-tac-tie game playing, development of ai languages, current trends in AI

UNIT-II:
Problem solving: state-space search and control strategies : Introduction, general problem solving, characteristics of problem, exhaustive searches, heuristic search techniques, iterative-deepening a*, constraint satisfaction
Problem reduction and game playing: Introduction, problem reduction, game playing, alpha-beta pruning, two-player perfect information games

UNIT-III:
Logic concepts: Introduction, propositional calculus, proportional logic, natural deduction system, axiomatic system, semantic tableau system in proportional logic, resolution refutation in proportional logic, predicate logic

UNIT-IV:
Knowledge representation: Introduction, approaches to knowledge representation, knowledge representation using semantic network, extended semantic networks for KR, knowledge representation using frames **advanced knowledge representation techniques:** Introduction, conceptual dependency theory, script structure, cyc theory, case grammars, semantic web

UNIT-V:
Expert system and applications: Introduction phases in building expert systems, expert system versus traditional systems, rule-based expert systems blackboard systems truth maintenance systems, application of expert systems, list of shells and tools

UNIT-VI:
Uncertainty measure: probability theory: Introduction, probability theory, Bayesian belief networks, certainty factor theory, dempster-shafer theory
Fuzzy sets and fuzzy logic: Introduction, fuzzy sets, fuzzy set operations, types of membership functions, multi valued logic, fuzzy logic, linguistic variables and hedges, fuzzy propositions, inference rules for fuzzy propositions, fuzzy systems.

OUTCOMES:
- Identify problems that are amenable to solution by AI methods, and which AI methods may be suited to solving a given problem.
- Formalize a given problem in the language/framework of different AI methods (e.g., as a search problem, as a constraint satisfaction problem, as a planning problem, as a Markov decision process, etc).
- Implement basic AI algorithms (e.g., standard search algorithms or dynamic programming).
- Design and carry out an empirical evaluation of different algorithms on problem formalization, and state the conclusions that the evaluation supports.

TEXT BOOKS:
1. Artificial Intelligence- Saroj Kaushik, CENGAGE Learning,
2. Artificial intelligence, A modern Approach, 2nded, Stuart Russel, Peter Norvig, PEA
3. Artificial Intelligence- Rich, Kevin Knight, Shiv Shankar B Nair, 3rded, TMH
4. Introduction to Artificial Intelligence, Patterson, PHI

REFERNCE BOOKS:
1. Atificial intelligence, structures and Strategies for Complex problem solving, -George F Lugar, 5thed, PEA
2. Introduction to Artificial Intelligence, Ertel, Wolf Gang, Springer
3. Artificial Intelligence, A new Synthesis, Nils J Nilsson, Elsevier

INTERNET OF THINGS
(Open Elective)

OBJECTIVES:
- Identify problems that are amenable to solution by AI methods, and which AI methods may be suited to solving a given problem.
- Formalize a given problem in the language/framework of different AI methods (e.g., as a search problem, as a constraint satisfaction problem, as a planning problem, as a Markov decision process, etc).
- Implement basic AI algorithms (e.g., standard search algorithms or dynamic programming).
- Design and carry out an empirical evaluation of different algorithms on problem formalization, and state the conclusions that the evaluation supports.

UNIT - I:

The Internet of Things: An Overview of Internet of things, Internet of Things Technology, behind IoTs Sources of the IoTs, M2M Communication, Examples OF IoTs, Design Principles For Connected Devices

UNIT – II:

Business Models for Business Processes in the Internet of Things ,IoT/M2M systems LAYERS AND designs standardizations ,Modified OSI Stack for the IoT/M2M Systems ,ETSI M2M domains and High-level capabilities ,Communication Technologies, Data Enrichment and Consolidation and Device Management Gateway Ease of designing and affordability

UNIT – III:

Design Principles for the Web Connectivity for connected-Devices, Web Communication protocols for Connected Devices, Message Communication protocols for Connected Devices, Web Connectivity for connected-Devices.

UNIT– IV:

Internet Connectivity Principles, Internet connectivity, Application Layer Protocols: HTTP, HTTPS, FTP, Telnet.

UNIT– V:

Data Acquiring, Organizing and Analytics in IoT/M2M, Applications/Services/Business Processes, IOT/M2M Data Acquiring and Storage, Business Models for Business Processes in

the Internet Of Things, Organizing Data, Transactions, Business Processes, Integration and Enterprise Systems.

UNIT – VI

Data Collection, Storage and Computing Using a Cloud Platform for IoT/M2M Applications/Services, Data Collection, Storage and Computing Using cloud platform Everything as a service and Cloud Service Models, IOT cloud-based services using the Xively (Pachube/COSM), Nimbits and other platforms Sensor, Participatory Sensing, Actuator, Radio Frequency Identification, and Wireless, Sensor Network Technology, Sensors Technology ,Sensing the World.

OUTCOMES:
- Demonstrate knowledge and understanding of the security and ethical issues of the Internet of Things
- Conceptually identify vulnerabilities, including recent attacks, involving the Internet of Things
- Develop critical thinking skills
- Compare and contrast the threat environment based on industry and/or device type

TEXTBOOKS:

- Internet of Things: Architecture, Design Principles And Applications, Rajkamal, McGraw Hill Higher Education
- Internet of Things, A.Bahgya and V.Madisetti, Univesity Press, 2015

REFERNCE BOOKS:

1. Designingthe Internet of Things, Adrian McEwen and Hakim Cassimally, Wiley
2. Getting Started with the Internet of Things CunoPfister , Oreilly

CYBER SECURITY
(Open Elective)

OBJECTIVES:
- The Cyber security Course will provide the students with foundational Cyber Security principles, Security architecture, risk management, attacks, incidents, and emerging IT and IS technologies.
- Students will gain insight into the importance of Cyber Security and the integral role of Cyber Security professionals.

UNIT- I: Introduction to Cybercrime:
Introduction, Cybercrime: Definition and Origins of the Word, Cybercrime and Information Security ,Who are Cybercriminals? , Classifications of Cybercrimes, Cybercrime: The Legal Perspectives, Cybercrimes: An Indian Perspective, Cybercrime and the Indian ITA 2000, A Global Perspective on Cybercrimes, Cybercrime Era: Survival Mantra for the Netizens

UNIT -II: Cyber offenses:
How Criminals Plan Them –Introduction, How Criminals Plan the Attacks, Social Engineering, Cyber stalking, Cyber cafe and Cybercrimes, Botnets: The Fuel for Cybercrime, Attack Vector Cloud Computing.

UNIT -III: Cybercrime Mobile and Wireless Devices:
Introduction, Proliferation of Mobile and Wireless Devices, Trends in Mobility, Credit Card Frauds in Mobile and Wireless Computing Era, Security Challenges Posed by Mobile Devices, Registry Settings for Mobile Devices, Authentication Service Security, Attacks on Mobile/Cell Phones, Mobile Devices: Security Implications for Organizations, Organizational Measures for Handling Mobile, Organizational Security Policies and Measures in Mobile Computing Era, Laptops.

UNIT -IV: Tools and Methods Used in Cybercrime:
Introduction, Proxy Servers and Anonymizers, Phishing, Password Cracking, Key loggers and Spywares, Virus and Worms, Trojan Horses and Backdoors, Steganography, DoS and DDoS Attacks, SQL Injection, Buffer Overflow, Attacks on Wireless Networks, Phishing and Identity Theft:Introduction,Phishing,IdentityTheft(IDTheft)

UNIT -V: Cybercrimes and Cyber security:
Why Do We Need Cyber laws: The Indian Context, The Indian IT Act, Challenges to Indian Law and Cybercrime Scenario in India, Consequences of Not Addressing the Weakness in Information Technology Act, Digital Signatures and the Indian IT Act, Information Security Planning and Governance, Information Security Policy Standards, Practices, The information Security Blueprint, Security education, Training and awareness program, Continuing Strategies.

UNIT -VI: Understanding Computer Forensics:

Introduction, Historical Background of Cyber forensics, Digital Forensics Science, The Need for Computer Forensics, Cyber forensics and Digital Evidence, Forensics Analysis of E-Mail, Digital Forensics Life Cycle, Chain of Custody Concept, Network Forensics, Approaching a Computer Forensics Investigation, Computer Forensics and Steganography, Relevance of the OSI 7 Layer Model to Computer Forensics, Forensics and Social Networking Sites: The Security/Privacy Threats, Computer Forensics from Compliance Perspective, Challenges in Computer Forensics, Special Tools and Techniques, Forensics Auditing, Antiforensics

OUTCOMES:
- Cyber Security architecture principles
- Identifying System and application security threats and vulnerabilities
- Identifying different classes of attacks
- Cyber Security incidents to apply appropriate response
- Describing risk management processes and practices
- Evaluation of decision making outcomes of Cyber Security scenarios

TEXT BOOKS:
1. Cyber Security: Understanding Cyber Crimes, Computer Forensics and Legal Perspectives, Nina Godbole, SunitBelapure, Wiley.
2. Principles of Information Security, MichealE.Whitman and Herbert J.Mattord, Cengage Learning.

REFERENCES:
1. Information Security, Mark Rhodes, Ousley, MGH.

DIGITAL SIGNAL PROCESSING
(Open Elective)

OBJECTIVES:
- To study DFT and its computation
- To study the design techniques for digital filters
- To study the finite word length effects in signal processing
- To study the non-parametric methods of power spectrum estimations
- To study the fundamentals of digital signal processors.

UNIT -I
Discrete Fourier Transform
DFT and its properties, Relation between DTFT and DFT, FFT computations using Decimation in time and Decimation in frequency algorithms, Overlap-add and save methods

UNIT -II
Infinite Impulse Response Digital Filters
Review of design of analogue Butterworth and Chebyshev Filters, Frequency transformation in analogue domain - Design of IIR digital filters using impulse invariance technique - Design of digital filters using bilinear transform - pre warping - Realization using direct, cascade and parallel   forms.

UNIT- III
Finite Impulse Response Digital Filters
Symmetric and Ant symmetric FIR filters - Linear phase FIR filters - Design using Hamming, Henning and Blackman Windows - Frequency sampling method - Realization of FIR filters - Transversal, Linear phase and Polyphasestructures.

UNIT -IV
Finite Word Length Effects
Fixed point and floating point number representations - Comparison - Truncation and Rounding errors - Quantization noise - derivation for quantization noise power - coefficient quantization error - Product quantization error –

UNIT -V
Overflow error - Round off noise power - limit cycle oscillations due to product round off and overflow errors - signal scaling

UNIT -VI
Multirate Signal Processing
Introduction to Multirate signal processing-Decimation-Interpolation-Polyphase implementation of FIR filters for interpolator and decimator -Multistage implementation of sampling rate conversion- Design of narrow band filters - Applications of Multirate signal processing.

OUTCOMES:
- an ability to apply knowledge of Mathematics, science, and engineering
- an ability to design and conduct experiments and interpret data
- an ability to design a system, component or process to meet desired needs within realistic constraints such as economic, environmental, social, political, ethical, health and safety, manufacturability, and sustainability
- an ability to function as part of a multi-disciplinary team

TEXT BOOKS:
1. John G Proakis and Manolakis, "Digital Signal Processing Principles, Algorithms and Applications", Pearson, Fourth Edition, 2007.
2. S.Salivahanan, A. Vallavaraj, C. Gnanapriya, Digital Signal Processing, TMH/McGraw HillInternational, 2007

REFERENCE BOOKS:
1. E.C. Ifeachor and B.W. Jervis, " Digital signal processing - A practical approach", Second edition, Pearson, 2002.
2. S.K. Mitra, Digital Signal Processing, A Computer Based approach, Tata Mc GrawHill, 1998.
3. P.P.Vaidyanathan, Multirate Systems & Filter Banks, Prentice Hall, Englewood cliffs, NJ, 1993.
4. Johny R. Johnson, Introduction to Digital Signal Processing, PHI, 2006.

.

EMBEDDED SYSTEMS
(Open Elective)

OBJECTIVES:
- Technology capabilities and limitations of the hardware, software components
- Methods to evaluate design tradeoffs between different technology choices.
- Design Methodologies

UNIT-I:
Introduction to Embedded systems: What is an embedded system Vs. General computing system, history, classification, major application areas, and purpose of embedded systems. Core of embedded system, memory, sensors and actuators, communication interface, embedded firmware, other system components, PCB and passive components.

UNIT-II:
8—bit microcontrollers architecture: Characteristics, quality attributes application specific, domain specific, embedded systems. Factors to be considered in selecting a controller, 8051 architecture, memory organization, registers, oscillator unit, ports, source current, sinking current, design examples.

UNIT-III:
RTOS and Scheduling, Operating basics, types, RTOS, tasks, process and threads, multiprocessing and multitasking, types of multitasking, non preemptive, preemptive scheduling.

UNIT-IV:
Task communication of RTOS, Shared memory, pipes, memory mapped objects, message passing, message queue, mailbox, signaling, RPC and sockets, task communication/synchronization issues, racing, deadlock, live lock, the dining philosopher's problem.

UNIT-V:
The producer-consumer problem, Reader writers problem, Priority Inversion, Priority ceiling, Task Synchronization techniques, busy waiting, sleep and wakery, semaphore, mutex, critical section objects, events, device, device drivers, how to clause an RTOS, Integration and testing of embedded hardware and fire ware.

UNIT-VI:
Simulators, emulators, Debuggers, Embedded Product Development life cycle (EDLC), Trends in embedded Industry, Introduction to ARM family of processor.

OUTCOMES:

Understand the basics of an embedded system
- Program an embedded system
- Design, implement and test an embedded system.

Identify the unique characteristics of real-time systems
- Explain the general structure of a real-time system
- Define the unique design problems and challenges of real-time systems

TEXT BOOK:
1. Introduction to embedded systems Shibu. K.V, TMH, 2009.

REFERENCE BOOKS:
1. Ayala &Gadre: The 8051 Microcontroller & Embedded Systems using Assembly and C, CENGAGE
2. Embedded Systems, Rajkamal, TMH, 2009.
3. Embedded Software Primer, David Simon, Pearson.
4. The 8051 Microcontroller and Embedded Systems, Mazidi, Mazidi, Pearson,.

ROBOTICS

(Open Elective)

OBJECTIVES:
- To introduce the basic concepts, parts of robots and types of robots.
- To make the student familiar with the various drive systems for robot, sensors and their applications in robots and programming of robots.
- To discuss about the various applications of robots, justification and implementation of robot.

UNIT- I:
Introduction
Specifications of Robots- Classifications of robots – Work envelope – Flexible automation versus Robotic technology – Applications of Robots ROBOT KINEMATICS AND DYNAMICS Positions,

UNIT-II:
Orientations and frames, Mappings
Changing descriptions from frame to frame, Operators: Translations, Rotations and Transformations - Transformation Arithmetic – D-H Representation - Forward and inverse Kinematics Of Six Degree of Freedom Robot Arm – Robot Arm dynamics

UNIT- III:
Robot Drives and Power Transmission Systems
Robot drive mechanisms, hydraulic – electric – servomotor- stepper motor - pneumatic drives, Mechanical transmission method - Gear transmission, Belt drives, cables, Roller chains, Link - Rod systems - Rotary-to-Rotary motion conversion, Rotary-to-Linear motion conversion, Rack and Pinion drives, Lead screws, Ball Bearing screws,

UNIT -IV:
Manipulators
Construction of Manipulators, Manipulator Dynamic and Force Control, Electronic and Pneumatic manipulators

UNIT- V:
Robot End Effectors
Classification of End effectors – Tools as end effectors. Drive system for grippers-Mechanical adhesive-vacuum-magnetic-grippers. Hooks&scoops. Gripper force analysis and gripper design. Active and passive grippers.

UNIT -VI:
Path planning & Programming

Trajectory planning and avoidance of obstacles, path planning, skew motion, joint integrated motion – straight line motion-Robot languages-computer control and Robot software.

OUTCOMES:
- The Student must be able to design automatic manufacturing cells with robotic control using
- The principle behind robotic drive system, end effectors, sensor, machine vision robot Kinematics and programming.

TEXT BOOKS:
1. Deb S. R. and Deb S., "Robotics Technology and Flexible Automation", Tata McGraw HillEducation Pvt. Ltd, 2010.
2. John J.Craig, "Introduction to Robotics", Pearson, 2009.
3. Mikell P. Grooveret. al., "Industrial Robots - Technology, Programming and Applications", McGraw Hill, New York, 2008.

REFERENCE BOOKS:
1. Richard D Klafter, Thomas A Chmielewski, Michael Negin, "Robotics Engineering – An Integrated Approach", Eastern Economy Edition, Prentice Hall of India Pvt. Ltd., 2006.
2. Fu K S, Gonzalez R C, Lee C.S.G, "Robotics: Control, Sensing, Vision and Intelligence", McGraw Hill, 1987

	L	T	P	C
III Year – II Semester	0	0	3	2

NETWORK PROGRAMMING LAB

OBJECTIVES:
- To write, execute and debug c programs which use Socket API.
- To understand the use of client/server architecture in application development
- To understand how to use TCP and UDP based sockets and their differences.
- To get acquainted with unix system internals like Socket files, IPC structures.
- To Design reliable servers using both TCP and UDP sockets

Prerequisites:

Knowledge of C Programming, Basic commands of UNIX.

List of Programs

1. Understanding and using of commands like ifconfig, netstat, ping, arp, telnet, ftp, finger, traceroute, whoisetc. Usage of elementary socket system calls (socket (), bind(), listen(), accept(),connect(),send(),recv(),sendto(),recvfrom()).

2. Implementation of Connection oriented concurrent service (TCP).

3. Implementation of Connectionless Iterative time service (UDP).

4. Implementation of Select system call.

5. Implementation of gesockopt (), setsockopt () system calls.

6. Implementation of getpeername () system call.

7. Implementation of remote command execution using socket system calls.

8. Implementation of Distance Vector Routing Algorithm.

9. Implementation of SMTP.

10. Implementation of FTP.

11. Implementation of HTTP.

12. Implementation of RSA algorithm.

Note: Implement programs 2 to 7 in C and 8 to 12 in JAVA.

OUTCOMES:
- Understand and explain the basic concepts of Grid Computing;
- Explain the advantages of using Grid Computing within a given environment;
- Prepare for any upcoming Grid deployments and be able to get started with a potentially available Grid setup.
- Discuss some of the enabling technologies e.g. high-speed links and storage area networks.
- Build computer grids.

SUGGESTED READING:

L	T	P	C
0	0	3	2

III Year – II Semester

SOFTWARE TESTING LAB

OBJECTIVES:
- Demonstrate the UML diagrams with ATM system descriptions.
- Demonstrate the working of software testing tools with c language.
- Study of testing tools- win runner, selenium etc.
- Writing test cases for various applications

1   Write programs in 'C' Language to demonstrate the working ofthe following constructs:

        i)   do...while

        ii)   while….do

        iii)   if…else

        iv)   switch

        v)   for

2   "A program written in 'C' language for Matrix Multiplication fails" Introspect the causes for its failure and write down the possible reasons for its failure.

3   Take any system (e.g. ATM system) and study its system specifications and report the various bugs.

4   Write the test cases for any known application (e.g. Banking application)

5   Create a test plan document for any application (e.g. Library Management System)

6   Study of Win Runner Testing Tool and its implementation

        a)   Win runner Testing Process and Win runner User Interface.

        b)   How Win Runner identifies GUI(Graphical User Interface) objects in an application and describes the two modesfor organizing GUI map files.

        c)   How to record a test script and explains the basics of Test Script Language (TSL).

        d)   How to synchronize a test when the application responds slowly.

        e)   How to create a test that checks GUI objects and compare the behaviour of GUI objects in different versions of the sample application.

        f)   How to create and run a test that checks bitmaps in your application and run the test on different versions ofthe sample application and examine any differences, pixel by pixel.

g) How to Create Data-Driven Tests which supports to run a single test on several sets of data from a data table.

h) How to read and check text found in GUI objects and bitmaps.

i) How to create a batch test that automatically runs the tests.

j) How to update the GUI object descriptions which in turn supports test scripts as the application changes.

7 Apply Win Runner testing tool implementation in any real time applications.

OUTCOMES:
- Find practical solutions to the problems
- Solve specific problems alone or in teams
- Manage a project from beginning to end
- Work independently as well as in teams

Define, formulate and analyze a problem

	L	T	P	C
III Year – II Semester	0	0	3	2

DATA WARE HOUSING AND DATA MINING LAB

OBJECTIVES:
- Practical exposure on implementation of well known data mining tasks.
- Exposure to real life data sets for analysis and prediction.
- Learning performance evaluation of data mining algorithms in a supervised and an unsupervised setting.
- Handling a small data mining project for a given practical domain.

System/Software Requirements:

- **Intel based desktop PC**
- **WEKA TOOL**

1. Demonstration of preprocessing on dataset student.arff

2. Demonstration of preprocessing on dataset labor.arff

3. Demonstration of Association rule process on dataset contactlenses.arff using apriori algorithm

4. Demonstration of Association rule process on dataset test.arff using apriori algorithm

5. Demonstration of classification rule process on dataset student.arff using j48 algorithm

6. Demonstration of classification rule process on dataset employee.arff using j48 algorithm

7. Demonstration of classification rule process on dataset employee.arff using id3 algorithm

8. Demonstration of classification rule process on dataset employee.arff using naïve bayes algorithm

9. Demonstration of clustering rule process on dataset iris.arff using simple k-means

10. Demonstration of clustering rule process on dataset student.arff using simple k- means.

OUTCOMES:

- The data mining process and important issues around data cleaning, pre-processing and integration.
- The principle algorithms and techniques used in data mining, such as clustering, association mining, classification and prediction..

III Year - II Semester

L	T	P	C
0	2	0	0

INTELLECTUAL PROPERTY RIGHTS AND PATENTS

Objectives:

*To know the importance of Intellectual property rights, which plays a vital role in advanced Technical and Scientific disciplines.

*Imparting IPR protections and regulations for further advancement, so that the students can familiarize with the latest developments.

Unit I: Introduction to Intellectual Property Rights (IPR)

Concept of Property - Introduction to IPR – International Instruments and IPR - WIPO - TRIPS – WTO -Laws Relating to IPR - IPR Tool Kit - Protection and Regulation - Copyrights and Neighboring Rights – Industrial Property – Patents - Agencies for IPR Registration – Traditional Knowledge –Emerging Areas of IPR - Layout Designs and Integrated Circuits – Use and Misuse of Intellectual Property Rights.

Unit II: Copyrights and Neighboring Rights

Introduction to Copyrights – Principles of Copyright Protection – Law Relating to Copyrights - Subject Matters of Copyright – Copyright Ownership – Transfer and Duration – Right to Prepare Derivative Works –Rights of Distribution – Rights of Performers **–** Copyright Registration – Limitations – Infringement of Copyright – Relief and Remedy – Case Law - Semiconductor Chip Protection Act.

UNIT III: Patents

Introduction to Patents - Laws Relating to Patents in India – Patent Requirements – Product Patent and Process Patent - Patent Search - Patent Registration and Granting of Patent - Exclusive Rights – Limitations - Ownership and Transfer —— Revocation of Patent – Patent Appellate Board - Infringement of Patent – Compulsory Licensing —— Patent Cooperation Treaty – New developments in Patents – Software Protection and Computer related Innovations.

UNIT IV: Trademarks

Introduction to Trademarks – Laws Relating to Trademarks – Functions of Trademark – Distinction between Trademark and Property Mark – Marks Covered under Trademark Law - Trade Mark Registration – Trade Mark Maintenance – Transfer of rights - Deceptive Similarities - Likelihood of Confusion - Dilution of Ownership – Trademarks Claims and Infringement – Remedies – Passing Off Action.

UNIT V: Trade Secrets

Introduction to Trade Secrets – General Principles - Laws Relating to Trade Secrets - Maintaining Trade Secret – Physical Security – Employee Access Limitation – Employee Confidentiality Agreements – Breach of Contract –Law of Unfair Competition – Trade Secret Litigation – Applying State Law.

UNIT VI: Cyber Law and Cyber Crime

Introduction to Cyber Law – Information Technology Act 2000 - Protection of Online and Computer Transactions - E-commerce - Data Security – Authentication and Confidentiality - Privacy - Digital Signatures – Certifying Authorities - Cyber Crimes - Prevention and Punishment – Liability of Network Providers.

- Relevant Cases Shall be dealt where ever necessary.

Outcome:

* IPR Laws and patents pave the way for innovative ideas which are instrumental for inventions to seek Patents.

*Student get an insight on Copyrights, Patents and Software patents which are instrumental for further advancements.

References:

1. Intellectual Property Rights (Patents & Cyber Law), Dr. A. Srinivas. Oxford University Press, New Delhi.
2. Deborah E.Bouchoux: Intellectual Property, Cengage Learning, New Delhi.
3. PrabhuddhaGanguli: Intellectual Property Rights, Tata Mc-Graw –Hill, New Delhi
4. Richard Stim: Intellectual Property, Cengage Learning, New Delhi.
5. Kompal Bansal &Parishit Bansal Fundamentals of IPR for Engineers, B. S. Publications (Press).
6. Cyber Law - Texts & Cases, South-Western's Special Topics Collections.
7. R.Radha Krishnan, S.Balasubramanian: Intellectual Property Rights, Excel Books. New Delhi.
8. M.Ashok Kumar and Mohd.Iqbal Ali: Intellectual Property Rights, Serials Pub.

	L	T	P	C
IV Year – I Semester

	L	T	P	C
	4	0	0	3

CRYPTOGRAPHY AND NETWORK SECURITY

OBJECTIVES:
- In this course the following principles and practice of cryptography and network security are covered:
- Classical systems, symmetric block ciphers (DES, AES, other contemporary symmetric ciphers)
- Public-key cryptography (RSA, discrete logarithms),
- Algorithms for factoring and discrete logarithms, cryptographic protocols, hash functions, authentication, key management, key exchange, signature schemes,
- Email and web security, viruses, firewalls, digital right management, and other topics.

UNIT- I:
Basic Principles
Security Goals, Cryptographic Attacks, Services and Mechanisms, Mathematics of Cryptography

UNIT- II:
Symmetric Encryption
Mathematics of Symmetric Key Cryptography, Introduction to Modern Symmetric Key Ciphers, Data Encryption Standard, Advanced Encryption Standard.

UNIT- III:
Asymmetric Encryption
Mathematics of Asymmetric Key Cryptography, Asymmetric Key Cryptography

UNIT- IV:
Data Integrity, Digital Signature Schemes & Key Management
Message Integrity and Message Authentication, Cryptographic Hash Functions, Digital Signature, Key Management.

UNIT -V:
Network Security-I
Security at application layer: PGP and S/MIME, Security at the Transport Layer: SSL and TLS

UNIT -VI:
Network Security-II
Security at the Network Layer: IPSec, System Security

OUTCOMES:
- To be familiarity with information security awareness and a clear understanding of its importance.
- To master fundamentals of secret and public cryptography
- To master protocols for security services
- To be familiar with network security threats and countermeasures
- To be familiar with network security designs using available secure solutions (such asPGP,
- SSL, IPSec, etc)

TEXT BOOKS:
1) Cryptography and Network Security, Behrouz A Forouzan, DebdeepMukhopadhyay, (3e) Mc Graw Hill.
2) Cryptography and Network Security, William Stallings, (6e) Pearson.
3) Everyday Cryptography, Keith M.Martin, Oxford.

REFERENCE BOOKS:
1) Network Security and Cryptography, Bernard Meneges, Cengage Learning.

L	T	P	C
4	0	0	3

IV Year – I Semester

SOFTWARE ARCHITECTURE AND DESIGN PATTERNS

OBJECTIVES:
The course should enable the student:

- To understand interrelationships, principles and guidelines governing architecture and evolution over time.
- To understand various architectural styles of software systems.
- To understand design patterns and their underlying object oriented concepts.
- To understand implementation of design patterns and providing solutions to real world software design problems.
- To understand patterns with each other and understanding the consequences of combining patterns on the overall quality of a system.

UNIT-I:
Envisioning Architecture
The Architecture Business Cycle, What is Software Architecture, Architectural patterns, reference models, reference architectures, architectural structures and views.

Creating and Architecture Quality Attributes, Achieving qualities, Architectural styles and patterns, designing the Architecture, Documenting software architectures, Reconstructing Software Architecture.

UNIT-II:
Analyzing Architectures
Architecture Evaluation, Architecture design decision making, ATAM, CBAM

Moving from One System to Many
Software Product Lines, Building systems from off the shelf components, Software architecture in future.

UNIT-III:
Patterns
Pattern Description, Organizing catalogs, role in solving design problems, Selection and usage.
Creational Patterns
Abstract factory, Builder, Factory method, Prototype, Singleton

UNIT-IV:
Structural Patterns
Adapter, Bridge, Composite, Decorator, Façade, Flyweight, PROXY.

UNIT-V:
Behavioral Patterns

Chain of responsibility, command, Interpreter, iterator, mediator, memento, observer, state, strategy, template method, visitor.

UNIT-VI:
Case Studies

A-7E – A case study in utilizing architectural structures, The World Wide Web - a case study in Interoperability, Air Traffic Control – a case study in designing for high availability, Celsius Tech – a case study in product line development.

A Case Study (Designing a Document Editor): Design Problems, Document Structure, Formatting, Embellishing the User Interface, Supporting Multiple Look-and-Feel Standards, Supporting Multiple Window Systems, User Operations, Spelling Checking and Hyphenation.

TEXT BOOKS:
1. Software Architecture in Practice, second edition, Len Bass, Paul Clements & Rick Kazman, Pearson Education, 2003.

2. Design Patterns, Erich Gamma, Pearson Education, 1995.

REFERENCE BOOKS:
1. Beyond Software architecture, Luke Hohmann, Addison wesley, 2003.
2. Software architecture, David M. Dikel, David Kane and James R. Wilson, Prentice Hall

   PTR,   2001

3. Software Design, David Budgen, second edition, Pearson education, 2003

4. Head First Design patterns, Eric Freeman & Elisabeth Freeman, O'REILLY, 2007.

5. Design Patterns in Java, Steven John Metsker & William C. Wake, Pearson education,

   2006

6. J2EE Patterns, Deepak Alur, John Crupi & Dan Malks, Pearson education, 2003.

7. Design Patterns in C#, Steven John metsker, Pearson education, 2004.

8. Pattern Oriented Software Architecture, F.Buschmann & others, John Wiley & Sons.

	L	T	P	C
IV Year – I Semester	4	0	0	3

WEB TECHNOLOGIES

OBJECTIVES:

- This course is designed to introduce students with no programming experience to the programming languages and techniques associated with the World Wide Web. The course will introduce web-based media-rich programming tools for creating interactive web pages.

UNIT-I: HTML, CSS

Basic Syntax, Standard HTML Document Structure, Basic Text Markup, Images, Hypertext Links, Lists, Tables, Forms, HTML5

CSS: Levels of Style Sheets, Style Specification Formats, Selector Forms, The Box Model, Conflict Resolution

UNIT-II:

Java script
The Basic of Java script: Objects, Primitives Operations and Expressions, Screen Output and Keyboard Input, Control Statements, Object Creation and Modification, Arrays, Functions, Constructors, Pattern Matching using Regular Expressions
DHTML: Positioning Moving and Changing Elements

UNIT-III:

XML: Document type Definition, XML schemas, Document object model, XSLT, DOM and SAX Approaches,

AJAX A New Approach: Introduction to AJAX, Integrating PHP and AJAX.

UNIT-IV:

PHP Programming: Introducing PHP: Creating PHP script, Running PHP script.
Working with variables and constants: Using variables, Using constants, Data types,Operators.**Controlling program flow:** Conditional statements,Control statements,Arrays,functions.Working with forms and Databases such as MySQL.

UNIT-V:

Introduction to PERL, Operators and if statements, Program design and control structures, Arrays, Hashs and File handling, Regular expressions, Subroutines, Retrieving documents from the web with Perl.

UNIT-VI:

Introduction to Ruby, Variables, types, simple I/O, Control, Arrays, Hashes, Methods, Classes, Iterators, Pattern Matching. Overview of Rails.

OUTCOMES:
- Analyze a web page and identify its elements and attributes.
- Create web pages using XHTML and Cascading Styles sheets.
- Build dynamic web pages.
- Build web applications using PHP.
- Programming through PERL and Ruby
- Write simple client-side scripts using AJAX

TEXT BOOKS:
1. Programming the World Wide Web, Robet W Sebesta, 7ed, Pearson.
2. Web Technologies, Uttam K Roy, Oxford
3. The Web Warrior Guide to Web Programming, Bai, Ekedahl, Farrelll, Gosselin, Zak, Karparhi, Maclntyre, Morrissey, Cengage

REFERENCE BOOKS:
1. Ruby on Rails Up and Running, Lightning fast Web development, Bruce Tate, Curt Hibbs, Oreilly (2006)
2. Programming Perl, 4ed, Tom Christiansen, Jonathan Orwant, Oreilly (2012)
3. Web Technologies, HTML< JavaScript, PHP, Java, JSP, XML and AJAX, Black book, Dream Tech.
4. An Introduction to Web Design, Programming, Paul S Wang, Sanda S Katila, Cengage Learning
5. http://www.upriss.org.uk/perl/PerlCourse.html

	L	T	P	C
IV Year – I Semester	4	0	0	3

MANAGERIAL ECONOMICS AND FINANCIAL ANALYSIS
(Common to all Branches)

- **Course Objectives:**
- The Learning objectives of this paper is to understand the concept and nature of Managerial Economics and its relationship with other disciplines and also to understand the Concept of Demand and Demand forecasting, Production function, Input Output relationship, Cost-Output relationship and Cost-Volume-Profit Analysis.
- To understand the nature of markets, Methods of Pricing in the different market structures and to know the different forms of Business organization and the concept of Business Cycles.
- To learn different Accounting Systems, preparation of Financial Statement and uses of different tools for performance evaluation. Finally, it is also to understand the concept of Capital, Capital Budgeting and the techniques used to evaluate Capital Budgeting proposals.

UNIT-I
Introduction to Managerial Economics and demand Analysis:
Definition of Managerial Economics –Scope of Managerial Economics and its relationship with other subjects –Concept of Demand, Types of Demand, Determinants of Demand- Demand schedule, Demand curve, Law of Demand and its limitations- Elasticity of Demand, Types of Elasticity of Demand and Measurement- Demand forecasting and Methods of forecasting, Concept of Supply and Law of Supply.

UNIT – II:
Production and Cost Analyses:
Concept of Production function- Cobb-Douglas Production function- Leontief production function - Law of Variable proportions-Isoquants and Isocosts and choice of least cost factor combination-Concepts of Returns to scale and Economies of scale-Different cost concepts: opportunity costs, explicit and implicit costs- Fixed costs, Variable Costs and Total costs –Cost – Volume-Profit analysis-Determination of Breakeven point(simple problems)-Managerial significance and limitations of Breakeven point.

UNIT – III:
Introduction to Markets, Theories of the Firm & Pricing Policies:
Market Structures: Perfect Competition, Monopoly, Monopolistic competition and Oligopoly – Features – Price and Output Determination – Managerial Theories of firm: Marris and Williamson's models – other Methods of Pricing: Average cost pricing, Limit Pricing, Market Skimming Pricing, Internet Pricing: (Flat Rate Pricing, Usage sensitive pricing) and Priority Pricing.

UNIT – IV:
Types of Business Organization and Business Cycles:
Features and Evaluation of Sole Trader, Partnership, Joint Stock Company – State/Public Enterprises and their forms – Business Cycles : Meaning and Features – Phases of a Business Cycle.

Unit – V:
Introduction to Accounting & Financing Analysis:
Introduction to Double Entry Systems – Preparation of Financial Statements-Analysis and Interpretation of Financial Statements-Ratio Analysis – Preparation of Funds flow and cash flow statements (Simple Problems)

UNIT – VI:
Capital and Capital Budgeting: Capital Budgeting: Meaning of Capital-Capitalization-Meaning of Capital Budgeting-Time value of money- Methods of appraising Project profitability: Traditional Methods(pay back period, accounting rate of return) and modern methods(Discounted cash flow method, Net Present Value method, Internal Rate of Return Method and Profitability Index)

Course Outcome:
*The Learner is equipped with the knowledge of estimating the Demand and demand elasticities for a product and the knowledge of understanding of the Input-Output-Cost relationships and estimation of the least cost combination of inputs.
* One is also ready to understand the nature of different markets and Price Output determination under various market conditions and also to have the knowledge of different Business Units.
*The Learner is able to prepare Financial Statements and the usage of various Accounting tools for Analysis and to evaluate various investment project proposals with the help of capital budgeting techniques for decision making.

TEXT BOOKS
1. Dr. N. AppaRao, Dr. P. Vijay Kumar: 'Managerial Economics and Financial Analysis', Cengage Publications, New Delhi – 2011
2. Dr. A. R. Aryasri – Managerial Economics and Financial Analysis, TMH 2011
3. Prof. J.V.Prabhakararao, Prof. P. Venkatarao. 'Managerial Economics and Financial Analysis', Ravindra Publication.

REFERENCES:

1.Dr. B. Kuberudu and Dr. T. V. Ramana: Managerial Economics & Financial Analysis, Himalaya Publishing House, 2014.
2. V. Maheswari: Managerial Economics, Sultan Chand.2014
3. Suma Damodaran: Managerial Economics, Oxford 2011.
4. VanithaAgarwal: Managerial Economics, Pearson Publications 2011.
5. Sanjay Dhameja: Financial Accounting for Managers, Pearson.
6. Maheswari: Financial Accounting, Vikas Publications.
7. S. A. Siddiqui& A. S. Siddiqui: Managerial Economics and Financial Analysis, New Age International Publishers, 2012
8. Ramesh Singh, Indian Economy, 7th Edn., TMH2015
9. Pankaj Tandon A Text Book of Microeconomic Theory, Sage Publishers, 2015
10. Shailaja Gajjala and Usha Munipalle, Univerties press, 2015

L	T	P	C
4	0	0	3

BIG DATA ANALYTICS
(Elective - 1)

OBJECTIVES:
- Optimize business decisions and create competitive advantage with Big Data analytics
- Introducing Java concepts required for developing map reduce programs
- Derive business benefit from unstructured data
- Imparting the architectural concepts of Hadoop and introducing map reduce paradigm
- To introduce programming tools PIG & HIVE in Hadoop echo system.

UNIT-I
Data structures in Java: Linked List, Stacks, Queues, Sets, Maps; Generics: Generic classes and Type parameters, Implementing Generic Types, Generic Methods, Wrapper Classes, Concept of Serialization

UNIT-II
Working with Big Data: Google File System, Hadoop Distributed File System (HDFS) – Building blocks of Hadoop (Namenode, Datanode, Secondary Namenode, JobTracker, TaskTracker), Introducing and Configuring Hadoop cluster (Local, Pseudo-distributed mode, Fully Distributed mode), Configuring XML files.

UNIT-III
Writing MapReduce Programs: A Weather Dataset, Understanding Hadoop API for MapReduce Framework (Old and New), Basic programs of Hadoop MapReduce: Driver code, Mapper code, Reducer code, RecordReader, Combiner, Partitioner

UNIT-IV
Hadoop I/O: The Writable Interface, WritableComparable and comparators, Writable Classes: Writable wrappers for Java primitives, Text, BytesWritable, NullWritable, ObjectWritable and GenericWritable, Writable collections, Implementing a Custom Writable: Implementing a RawComparator for speed, Custom comparators

UNIT-V
Pig: Hadoop Programming Made Easier
Admiring the Pig Architecture, Going with the Pig Latin Application Flow, Working through the ABCs of Pig Latin, Evaluating Local and Distributed Modes of Running Pig Scripts, Checking out the Pig Script Interfaces, Scripting with Pig Latin

UNIT-VI
Applying Structure to Hadoop Data with Hive:
Saying Hello to Hive, Seeing How the Hive is Put Together, Getting Started with Apache Hive, Examining the Hive Clients, Working with Hive Data Types, Creating and Managing Databases and Tables, Seeing How the Hive Data Manipulation Language Works, Querying and Analyzing Data

OUTCOMES:

- Preparing for data summarization, query, and analysis.
- Applying data modeling techniques to large data sets
- Creating applications for Big Data analytics
- Building a complete business data analytic solution

TEXT BOOKS:

1. Big Java 4th Edition, Cay Horstmann, Wiley John Wiley & Sons, INC
2. Hadoop: The Definitive Guide by Tom White, 3rd Edition, O'reilly
3. Hadoop in Action by Chuck Lam, MANNING Publ.
4. Hadoop for Dummies by Dirk deRoos, Paul C.Zikopoulos, Roman B.Melnyk,Bruce Brown, Rafael Coss

REFERENCE BOOKS:

1. Hadoop in Practice by Alex Holmes, MANNING Publ.
2. Hadoop MapReduce Cookbook, SrinathPerera, ThilinaGunarathne

SOFTWARE LINKS:

1. Hadoop:http://hadoop.apache.org/
2. Hive: https://cwiki.apache.org/confluence/display/Hive/Home
3. Piglatin: http://pig.apache.org/docs/r0.7.0/tutorial.html

INFORMATION RETRIEVAL SYSTEMS
(Elective - 1)

OBJECTIVES
- To provide the foundation knowledge in information retrieval.
- To equip students with sound skills to solve computational search problems.
- To appreciate how to evaluate search engines.
- To appreciate the different applications of information retrieval techniques in the Internet or Web environment.
- To provide hands-on experience in building search engines and/or hands-on experience in evaluating search engines.

UNIT - I:
Introduction to Information Storage and Retrieval System: Introduction, Domain Analysis of IR systems and other types of Information Systems, IR System Evaluation.
Introduction to Data Structures and Algorithms related to Information Retrieval: Basic Concepts, Data structures, Algorithms

UNIT- II:
Inverted files: Introduction, Structures used in Inverted Files, Building Inverted file using a sorted array, Modifications to Basic Techniques.

UNIT -III:
Signature Files: Introduction, Concepts of Signature Files, Compression, Vertical Partitioning, Horizontal Partitioning.

UNIT- IV:
New Indices for Text: PAT Trees and PAT Arrays: Introduction, PAT Tree structure, algorithms on the PAT Trees, Building PAT trees as PATRICA Trees, PAT representation as arrays.

UNIT- V:
Stemming Algorithms: Introduction, Types of Stemming Algorithms, Experimental Evaluations of Stemming to Compress Inverted Files

UNIT- VI:
Thesaurus Construction: Introduction, Features of Thesauri, Thesaurus Construction, Thesaurus construction from Texts, Merging existing Thesauri

OUTCOMES
- Identify basic theories in information retrieval systems
- Identify the analysis tools as they apply to information retrieval systems
- Understands the problems solved in current IR systems
- Describes the advantages of current IR systems
- Understand the difficulty of representing and retrieving documents.
- Understand the latest technologies for linking, describing and searching the web.

TEXT BOOK:
1. Frakes, W.B., Ricardo Baeza-Yates: Information Retrieval Data Structures and Algorithms, Prentice Hall, 1992.
2. Modern Information Retrieval by Yates Pearson Education.
3. Information Storage & Retrieval by Robert Korfhage – John Wiley & Sons.

REFERENCES:
1. Kowalski, Gerald, Mark T Maybury: Information Retrieval Systems: Theory and Implementation, Kluwer Academic Press, 1997.
2. Information retrieval Algorithms and Heuristics, 2ed, Springer

MOBILE COMPUTING
(Elective - 1)

OBJECTIVE:

- To make the student understand the concept of mobile computing paradigm, its novel applications and limitations.
- To understand the typical mobile networking infrastructure through a popular GSM protocol
- To understand the issues and solutions of various layers of mobile networks, namely MAC layer, Network Layer & Transport Layer
- To understand the database issues in mobile environments & data delivery models.
- To understand the ad hoc networks and related concepts.
- To understand the platforms and protocols used in mobile environment.

UNIT- I
Introduction: Mobile Communications, Mobile Computing – Paradigm, Promises/Novel Applications and Impediments and Architecture; Mobile and Handheld Devices, Limitations of Mobile and Handheld Devices.
GSM – Services, System Architecture, Radio Interfaces, Protocols, Localization, Calling, Handover, Security, New Data Services, GPRS.

UNIT –II
(Wireless) Medium Access Control (MAC) : Motivation for a specialized MAC (Hidden and exposed terminals, Near and far terminals), SDMA, FDMA, TDMA, CDMA, Wireless LAN/(IEEE 802.11)

UNIT –III
Mobile Network Layer: IP and Mobile IP Network Layers, Packet Delivery and Handover Management, Location Management, Registration, Tunneling and Encapsulation, Route Optimization, DHCP.

UNIT –IV
Mobile Transport Layer: Conventional TCP/IP Protocols, Indirect TCP, Snooping TCP, Mobile TCP, Other Transport Layer Protocols for Mobile Networks.
Database Issues: Database Hoarding & Caching Techniques, Client-Server Computing & Adaptation, Transactional Models, Query processing, Data Recovery Process &QoS Issues.

UNIT- V
Data Dissemination and Synchronization : Communications Asymmetry, Classification of Data Delivery Mechanisms, Data Dissemination, Broadcast Models, Selective Tuning and Indexing Methods, Data Synchronization – Introduction, Software, and Protocols.

UNIT- VI

Mobile Ad hoc Networks (MANETs) : Introduction, Applications & Challenges of a MANET, Routing, Classification of Routing Algorithms, Algorithms such as DSR, AODV, DSDV, etc. , Mobile Agents, Service Discovery.

Protocols and Platforms for Mobile Computing : WAP, Bluetooth, XML, J2ME, Java Card, PalmOS, Windows CE, SymbianOS, Linux for Mobile Devices, Android.

OUTCOMES:
- Able to think and develop new mobile application.
- Able to take any new technical issue related to this new paradigm and come up with a solution(s).
- Able to develop new ad hoc network applications and/or algorithms/protocols.
- Able to understand & develop any existing or new protocol related to mobile environment

TEXT BOOKS:
1. Jochen Schiller, "Mobile Communications", Addison-Wesley, Second Edition, 2009.
2. Raj Kamal, "Mobile Computing", Oxford University Press, 2007, ISBN: 0195686772

REFERENCE BOOKS:
1. ASOKE K TALUKDER, HASAN AHMED, ROOPA R YAVAGAL, "Mobile Computing, Technology Applications and Service Creation" Second Edition, Mc Graw Hill.
2. UWE Hansmann, LotherMerk, Martin S. Nocklous, Thomas Stober, "Principles of Mobile Computing," Second Edition, Springer.

L	T	P	C
4	0	0	3

CLOUD COMPUTING
(Elective - 2)

OBJECTIVES:
- The student will learn about the cloud environment, building software systems and components that scale to millions of users in modern internetcloud concepts capabilities across the various cloud service models including Iaas, Paas, Saas, and developing cloud based software applications on top of cloud platforms.

UNIT -I: Systems modeling, Clustering and virtualization
Scalable Computing over the Internet, Technologies for Network based systems, System models for Distributed and Cloud Computing, Software environments for distributed systems and clouds, Performance, Security And Energy Efficiency

UNIT- II:Virtual Machines and Virtualization of Clusters and Data Centers
Implementation Levels of Virtualization, Virtualization Structures/ Tools and mechanisms, Virtualization of CPU, Memory and I/O Devices, Virtual Clusters and Resource Management, Virtualization for Data Center Automation.

UNIT- III: Cloud Platform Architecture
Cloud Computing and service Models, Architectural Design of Compute and Storage Clouds, Public Cloud Platforms, Inter Cloud Resource Management, Cloud Security and Trust Management. Service Oriented Architecture, Message Oriented Middleware.

UNIT -IV: Cloud Programming and Software Environments
Features of Cloud and Grid Platforms, Parallel & Distributed Programming Paradigms, Programming Support of Google App Engine, Programming on Amazon AWS and Microsoft Azure, Emerging Cloud Software Environments.

UNIT- V: Cloud Resource Management and Scheduling
Policies and Mechanisms for Resource Management Applications of Control Theory to Task Scheduling on a Cloud, Stability of a Two Level Resource Allocation Architecture, Feedback Control Based on Dynamic Thresholds. Coordination of Specialized Autonomic Performance Managers, Resource Bundling, Scheduling Algorithms for Computing Clouds, Fair Queuing, Start Time Fair Queuing, Borrowed Virtual Time, Cloud Scheduling Subject to Deadlines, Scheduling MapReduce Applications Subject to Deadlines.

UNIT- VI: Storage Systems
Evolution of storage technology, storage models, file systems and database, distributed file systems, general parallel file systems. Google file system. Apache Hadoop, Big Table, Megastore, Amazon Simple Storage Service (S3)

OUTCOMES:
- Understanding the key dimensions of the challenge of Cloud Computing
- Assessment of the economics , financial, and technological implications for selecting cloud computing for own organization
- Assessing the financial, technological, and organizational capacity of employer's for actively initiating and installing cloud-based applications.
- Assessment of own organizations' needs for capacity building and training in cloud computing-related IT areas

TEXT BOOKS:
1. Distributed and Cloud Computing, Kai Hwang, Geoffry C. Fox, Jack J. Dongarra MK Elsevier.
2. Cloud Computing, Theory and Practice, Dan C Marinescu, MK Elsevier.
3. Cloud Computing, A Hands on approach, ArshadeepBahga, Vijay Madisetti, University Press

REFERNCE BOOKS:
1. Cloud Computing, A Practical Approach, Anthony T Velte, Toby J Velte, Robert Elsenpeter, TMH
2. Mastering Cloud Computing, Foundations and Application Programming, Raj Kumar Buyya, Christen vecctiola, S Tammaraiselvi, TMH

SOFTWARE PROJECT MANAGEMENT
(Elective - 2)

OBJECTIVES:
- To study how to plan and manage projects at each stage of the software development life cycle (SDLC)
- To train software project managers and other individuals involved in software project planning and tracking and oversight in the implementation of the software project management process.
- To understand successful software projects that support organization's strategic goals

UNIT -I:Introduction
Project, Management, Software Project Management activities, Challenges in software projects, Stakeholders, Objectives & goals
Project Planning: Step-wise planning, Project Scope, Project Products & deliverables, Project activities, Effort estimation, Infrastructure

UNIT -II:Project Approach
Lifecycle models, Choosing Technology, Prototyping
Iterative & incremental Process Framework; Lifecycle phases, Process Artifacts, Process workflows (Book 2)

UNIT -III:Effort estimation & activity Planning
Estimation techniques, Function Point analysis, SLOC, COCOMO, Use case-based estimation , Activity Identification Approaches, Network planning models, Critical path analysis

UNIT -IV: Risk Management
Risk categories, Identification, Assessment, Planning and management, PERT technique, Monte Carlo approach

UNIT -V:Project Monitoring & Control, Resource Allocation
Creating a framework for monitoring & control, Progress monitoring, Cost monitoring, Earned value Analysis, Defects Tracking, Issues Tracking, Status reports, Types of Resources, Identifying resource requirements, Resource scheduling

UNIT -VI:Software Quality
Planning Quality, Defining Quality - ISO 9016, Quality Measures, Quantitative Quality Management Planning, Product Quality & Process Quality
 Metrics, Statistical Process Control Capability Maturity Model, Enhancing software Quality (Book3)

OUTCOMES:

- To match organizational needs to the most effective software development model
- To understand the basic concepts and issues of software project management
- To effectively Planning the software projects
- To implement the project plans through managing people, communications and change
- To select and employ mechanisms for tracking the software projects
- To conduct activities necessary to successfully complete and close the Software projects
- To develop the skills for tracking and controlling software deliverables
- To create project plans that address real-world management challenges

TEXT BOOKS:
1. Software Project Management, Bob Hughes & Mike Cotterell, TATA Mcgraw-Hill
2. Software Project Management, Walker Royce: Pearson Education, 2005.
3. Software Project Management in practice, Pankaj Jalote, Pearson.

REFERENCE BOOKS:
1. Software Project Management, Joel Henry, Pearson Education.

SCRIPTING LANGUAGES
(Elective - 2)

OBJECTIVES:
- The course demonstrates an in depth understanding of the tools and the scripting languages necessary for design and development of applications dealing with Bio-information/ Bio-data.
- The instructor is advised to discuss examples in the context of Bio-data/ Bio-information application development.

UNIT - I
Introduction to PERL and Scripting Scripts and Programs, Origin of Scripting , Scripting Today, Characteristics of Scripting Languages, Uses for Scripting Languages, Web Scripting, and the universe of Scripting Languages. PERL- Names and Values, Variables, Scalar Expressions, Control Structures, arrays, list, hashes, strings, pattern and regular expressions, subroutines.

UNIT - II
Advanced perl Finer points of looping, pack and unpack, file system, eval, data structures, packages, modules, objects, interfacing to the operating system, Creating Internet ware applications, Dirty Hands Internet Programming, security Issues.

UNIT- III
PHP Basics PHP Basics- Features, Embedding PHP Code in your Web pages, Outputting the data to the browser, Data types, Variables, Constants, expressions, string interpolation, control structures, Function, Creating a Function, Function Libraries, Arrays, strings and Regular Expressions.

UNIT - IV
Advanced PHP Programming PHP and Web Forms, Files, PHP Authentication and Methodologies -Hard Coded, File Based, Database Based, IP Based, Login Administration, Uploading Files with PHP, Sending Email using PHP, PHP Encryption Functions, the Mcrypt package, Building Web sites for the World.

UNIT -V
TCL Structure, syntax, Variables and Data in TCL, Control Flow, Data Structures, input/output, procedures , strings , patterns, files, Advance TCL- eval, source, exec and uplevel commands, Name spaces, trapping errors, event driven programs, making applications internet aware, Nuts and Bolts Internet Programming, Security Issues, C Interface. Tk-Visual Tool Kits, Fundamental Concepts of Tk, Tk by example, Events and Binding , Perl-Tk.

UNIT- VI
Python Introduction to Python language, python-syntax, statements, functions, Built-in-functions and Methods, Modules in python, Exception Handling. Integrated Web Applications in Python – Building Small, Efficient Python Web Systems, Web Application Framework.

OUTCOMES:

- To master the theory behind scripting and its relationship to classic programming.
- To survey many of the modern and way cool language features that show up frequently in scripting languages.
- To gain some fluency programming in Ruby, JavaScript, Perl, Python, and related languages.
- To design and implement one's own scripting language.

TEXT BOOKS:

1. The World of Scripting Languages, David Barron, Wiley Publications.
2. Python Web Programming, Steve Holden and David Beazley, New Riders Publications.
3. Beginning PHP and MySQL, 3rd Edition, Jason Gilmore, Apress Publications (Dream tech)

REFERENCE BOOKS:

1. Open Source Web Development with LAMP using Linux, Apache, MySQL, Perl and PHP, J.Lee and B.Ware (Addison Wesley) Pearson Education.
2. Programming Python, M.Lutz,SPD.
3. PHP 6 Fast and Easy Web Development, Julie Meloni and Matt Telles, Cengage Learning Publications.
4. PHP 5.1, I.Bayross and S.Shah, The X Team, SPD.
5. Core Python Programming, Chun, Pearson Education.
6. Guide to Programming with Python, M.Dawson, Cengage Learning.
7. Perl by Example, E.Quigley, Pearson Education.

L	T	P	C
0	0	3	2

SOFTWARE ARCHITECTURE AND DESIGN PATTERNS LAB

Software Architecture Lab

The course project is divided in 6 small components that will be performed during the different lab sessions; there are, in principle, 7 lab sessions. The project consists of the design and implementation of the software architecture of a Weather Mapping System (WMS). Implementation will take place both in Java and C++ (combination of both languages). Each lab assignment consists of a theoretical part and a practical part, which are defined in specific lab assignment statements that are posted at least one or two weeks before the session.

Report and demo (if applicable) for each assignment is due for the following session.

1. Tool Presentation

This session is an introductory session; there is no lab assignment for this session.

Introduction to working with an industrial strength software development environment, namely Rational Rose: how to write and maintain a UML specification; configuration management; architecture design; CORBA-IDL document generation; Java code generation from a UML model etc.

Presentation of the Project: Weather Mapping System.

2. Use Case View
Design of the Use Case View. Risk Analysis.

3: Logical View

Design of the Logical View of the Weather Mapping System (WMS).

4: Integrating Patterns in the Architecture
Integration of selected architectural and design patterns in the logical view obtained previously.

5: Implementation, Process, and Deployment Views

Design of the implementation, process, and deployment views for the Weather Mapping System.

6: Component and Interprocess Communication Design

Generation from the previous architecture design of CORBA Interfaces and Components Definitions.

7: Implementation of WMS

Implementation of the Weather Mapping System (Java & C++), with a particular emphasis on the Interprocess communication mechanism and the software components identified.

Lab Reports:

Lab reports should include:

- The answers to the questions included in the assignment statement. The answers should motivate briefly your design choices.
- The printout of the diagrams and related documents (e.g. class, use cases, operations descriptions etc.) produced using Rational Rose.

Reference: http://www.ece.uvic.ca/~itraore/seng422-06/eng422-06.html

Design Patterns Lab

S. No	Programs
1.	Use case Diagram for Librarian Scenario
2.	Using UML design Abstract factory design pattern
3.	Using UML design Adapter-class Design pattern
4.	Using UML design Adapter-object Design pattern
5.	Using UML design Strategy Design pattern
6.	Using UML design Builder Design pattern
7.	Using UML design Bridge Design pattern
8.	Using UML design Decorator Design pattern
9.	User gives a print command from a word document. Design to represent this chain of responsibility Design pattern
10.	Design a Flyweight Design pattern
11.	Using UML design Facade Design pattern

12. Using UML design Iterator Design pattern   .

13. Using UML design Mediator Design pattern

14. Using UML design Proxy Design pattern

15. Using UML design Visitor Design pattern

L	T	P	C
0	0	3	2

WEB TECHNOLOGIES LAB

OBJECTIVES:
- To acquire knowledge of XHTML, Java Script and XML to develop web applications
- Ability to develop dynamic web content using Java Servlets and JSP
- To understand JDBC connections and Java Mail API
- To understand the design and development process of a complete web application

1. Design the following static web pages required for an online book store web site.

1) HOME PAGE:
The static home page must contain three **frames**.

Top frame: Logo and the college name and links to Home page, Login page, Registration page, Catalogue page and Cart page (the description of these pages will be given below).

Left frame: At least four links for navigation, which will display the catalogue of respective links.

For e.g.: When you click the link **"MCA"** the catalogue for MCABooks should be displayed in the Right frame.

Right frame: The *pages to the links in the left frame must be loaded here*. Initially this page contains description of the web site.

Logo	Web Site Name			
Home	Login	Registration	Catalogue	Cart
mca mba BCA	Description of the Web Site			
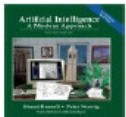
2) login page

Logo	Web Site Name			
Home	Login	Registration	Catalogue	Cart
MCA MBA BCA	Login : `11a51f0003` Password: `*******` Submit Reset			

3) **CATOLOGUE PAGE:**
The catalogue page should contain the details of all the books available in the web site in a table.
The details should contain the following:
1. Snap shot of Cover Page.
2. Author Name.
3. Publisher.
4. Price.
5. Add to cart button.

4. **REGISTRATION PAGE:**
Create a "*registration form* "with the following fields
1) Name (Text field)
2) Password (password field)
3) E-mail id (text field)
4) Phone number (text field)
5) Sex (radio button)
6) Date of birth (3 select boxes)
7) Languages known (check boxes – English, Telugu, Hindi, Tamil)
8) Address (text area)

5. **Design a web page using CSS (C**ascading **S**tyle **S**heets**)** which includes the following:
1) Use different font, styles:
In the style definition you define how each selector should work (font, color etc.).
Then, in the body of your pages, you refer to these selectors to activate the styles

6. **Write an XML** file which will display the Book information which includes the following:
1) Title of the book
2) Author Name
3) ISBN number
4) Publisher name
5) Edition
6) Price

Write a Document Type Definition (DTD) to validate the above XML file.

7. Write Ruby program reads a number and calculates the factorial value of it and prints the Same.

8. Write a Ruby program which counts number of lines in a text files using its regular Expressions facility.

9. Write a Ruby program that uses iterator to find out the length of a string.

10. Write simple Ruby programs that uses arrays in Ruby.

11. Write programs which uses associative arrays concept of Ruby.

12. Write Ruby program which uses Math module to find area of a triangle.

13. Write Ruby program which uses tk module to display a window

14. Define complex class in Ruby and do write methods to carry operations on complex objects.

15. Write a program which illustrates the use of associative arrays in perl.

16. Write perl program takes set names along the command line and prints whether they are regular files or special files

17. Write a perl program to implement UNIX `passed' program

18. An example perl program to connect to a MySQl database table and executing simple commands.

19. Example PHP program for cotactus page.

20. User Authentication:
Assume four users user1, user2, user3 and user4 having the passwords pwd1, pwd2, pwd3 and pwd4 respectively. Write a PHP for doing the following.
1. Create a Cookie and add these four user id's and passwords to this Cookie.
2. Read the user id and passwords entered in the Login form (week1)   and authenticate with the values (user id and passwords) available in the cookies.
If he is a valid user (i.e., user-name and password match) you should welcome him by name (user-name) else you should display "You are not an authenticated user ''.
Use init-parameters to do this.

21. Example PHP program for registering users of a website and login.

22. Install a database(Mysql or Oracle).
Create a table which should contain at least the following fields: name, password, email-id, phone number(these should hold the data from the registration form).
Write a PHP program   to connect to that database and extract data from the tables and display them. Experiment with various SQL queries.
Insert the details of the users who register with the web site, whenever a new user clicks the submit button in the registration page (week2).

23. Write a PHP which does the following job:
Insert the details of the 3 or 4 users who register with the web site (week9) by using registration form. Authenticate the user when he submits the login form using the user name and password from the database (similar to week8 instead of cookies).

24.Create tables in the database which contain the details of items (books in our case like Book name , Price, Quantity, Amount) of each category. Modify your catalogue page (week 2)in such a way that you should connect to the database and extract data from the tables and display them in the catalogue page using PHP

25.HTTP is a stateless protocol. Session is required to maintain the state.
The user may add some items to cart from the catalog page. He can check the cart page for the selected items. He may visit the catalogue again and select some more items. Here our interest is the selected items should be added to the old cart rather than a new cart. Multiple users can do the same thing at a time(i.e., from different systems in the LAN using the ip-address instead of local host). This can be achieved through the use of sessions. Every user will have his own session which will be created after his successful login to the website. When the user logs out his session should get invalidated (by using the method session. Invalidate ().
Modify your catalogue and cart PHP pages to achieve the above mentioned functionality using sessions.

OUTCOMES:
- Students will be able to develop static web sites using XHTML and Java Scripts
- To implement XML and XSLT for web applications
- Develop Dynamic web content using Java Servlets and JSP
- To develop JDBC connections and implement a complete Dynamic web application

	L	T	P	C
IV Year – II Semester	4	0	0	3

DISTRIBUTED SYSTEMS

OBJECTIVES:
- Provides an introduction to the fundamentals of distributed computer systems, assuming the availability of facilities for data transmission, IPC mechanisms in distributed systems, Remote procedure calls.
- Expose students to current technology used to build architectures to enhance distributed Computing infrastructures with various computing principles

UNIT-I:
Characterization of Distributed Systems: Introduction, Examples of Distributed Systems, Resource Sharing and the Web, Challenges.
System Models: Introduction, Architectural Models- Software Layers, System Architecture, Variations, Interface and Objects, Design Requirements for Distributed Architectures, Fundamental Models- Interaction Model, Failure Model, Security Model.

UNIT-II:
Interprocess Communication: Introduction, The API for the Internet Protocols- The Characteristics of Interprocess communication, Sockets, UDP Datagram Communication, TCP Stream Communication; External Data Representation and Marshalling; Client Server Communication; Group Communication- IP Multicast- an implementation of group communication, Reliability and Ordering of Multicast.

UNIT-III:
Distributed Objects and Remote Invocation: Introduction, Communication between Distributed Objects- Object Model, Distributed Object Modal, Design Issues for RMI, Implementation of RMI, Distributed Garbage Collection; Remote Procedure Call, Events and Notifications, Case Study: JAVA RMI.

UNIT-IV:
Operating System Support: Introduction, The Operating System Layer, Protection, Processes and Threads –Address Space, Creation of a New Process, Threads.

UNIT-V:
Distributed File Systems: Introduction, File Service Architecture; Peer-to-Peer Systems: Introduction, Napster and its Legacy, Peer-to-Peer Middleware, Routing Overlays.
Coordination and Agreement: Introduction, Distributed Mutual Exclusion, Elections, Multicast Communication.

UNIT-VI:
Transactions & Replications: Introduction, System Model and Group Communication, Concurrency Control in Distributed Transactions, Distributed Dead Locks, Transaction Recovery; Replication-Introduction, Passive (Primary) Replication, Active Replication.

OUTCOMES:
- Develop a familiarity with distributed file systems.
- Describe important characteristics of distributed systems and the salient architectural features of such systems.
- Describe the features and applications of important standard protocols which are used in distributed systems.
- Gaining practical experience of inter-process communication in a distributed environment

TEXT BOOKS:
1. Ajay D Kshemkalyani, MukeshSighal, "Distributed Computing, Principles, Algorithms and Systems", Cambridge
2. George Coulouris, Jean Dollimore, Tim Kindberg, "Distributed Systems- Concepts and Design", Fourth Edition, Pearson Publication

REFERENCE BOOKS
1. Distributed-Systems-Principles-Paradigms-Tanenbaum PHI

	L	**T**	**P**	**C**
IV Year – II Semester	**4**	**0**	**0**	**3**

MANAGEMENT SCIENCE

Course Objectives:
*To familiarize with the process of management and to provide basic insight into select contemporary management practices
*To provide conceptual knowledge on functional management and strategic management.

UNIT I
Introduction to Management: Concept –nature and importance of Management –Generic Functions of Management – Evaluation of Management thought– Theories of Motivation – Decision making process-Designing organization structure- Principles of organization – Organizational typology- International Management: Global Leadership and Organizational behavior Effectiveness(GLOBE) structure

UNIT II
Operations Management: Principles and Types of Management – Work study- Statistical Quality Control- Control charts (P-chart, R-chart, and C-chart) Simple problems- Material Management: Need for Inventory control- EOQ, ABC analysis (simple problems) and Types of ABC analysis (HML, SDE, VED, and FSN analysis).

UNIT III
Functional Management: Concept of HRM, HRD and PMIR- Functions of HR Manager- Wage payment plans(Simple Problems) – Job Evaluation and Merit Rating - Marketing Management- Functions of Marketing – Marketing strategies based on product Life Cycle, Channels of distributions. Operationlizing change through performance management.

UNIT IV
Project Management: (PERT/CPM): Development of Network – Difference between PERT and CPM Identifying Critical Path- Probability- Project Crashing (Simple Problems)

Unit V
Strategic Management: Vision, Mission, Goals, Strategy – Elements of Corporate Planning Process – Environmental Scanning – SWOT analysis- Steps in Strategy Formulation and Implementation, Generic Strategy Alternatives. Global strategies, theories of Multinational Companies.

UNIT VI
Contemporary Management Practice: Basic concepts of MIS, MRP, Justin- Time(JIT) system, Total Quality Management(TQM), Six sigma and Capability Maturity Model(CMM) Levies, Supply Chain Management , Enterprise Resource Planning (ERP), Business Process outsourcing (BPO), Business process Re-engineering and Bench Marking, Balanced Score Card.

Course Outcome:
*After completion of the Course the student will acquire the knowledge on management functions, global leadership and organizational behavior.
*Will familiarize with the concepts of functional management project management and strategic management.

Text Books
1. Dr. P. Vijaya Kumar & Dr. N. Appa Rao, '*Management Science'* Cengage, Delhi, 2012.
2. Dr. A. R. Aryasri, *Management Science'* TMH 2011.

References:

1. Koontz & Weihrich: 'Essentials *of management'* TMH 2011
2. Seth & Rastogi: Global Management Systems, Cengage learning , Delhi, 2011
3. Robbins: Organizational Behaviour, Pearson publications, 2011
4. Kanishka Bedi: Production & Operations Management, Oxford   Publications, 2011
5. Philip Kotler & Armstrong: Principles of Marketing, Pearson publications
6. Biswajit Patnaik: Human Resource Management, PHI, 2011
7. Hitt and Vijaya Kumar: Starategic Management, Cengage learning
8. Prem Chadha: Performance Management, Trinity Press(An Imprint of Laxmi Publications Pvt. Ltd.)   Delhi 2015.
9. Anil Bhat& Arya Kumar : Principles of Management, Oxford University Press, New Delhi, 2015.

	L	T	P	C
	4	0	0	3

IV Year – II Semester

MACHINE LEARNING

OBJECTIVES:
- Familiarity with a set of well-known supervised, unsupervised and semi-supervised
- learning algorithms.
- The ability to implement some basic machine learning algorithms
- Understanding of how machine learning algorithms are evaluated

UNIT -I:The ingredients of machine learning, Tasks: the problems that can be solved with machine learning, Models: the output of machine learning, Features, the workhorses of machine learning. **Binary classification and related tasks:** Classification, Scoring and ranking, Class probability estimation

UNIT- II:Beyond binary classification: Handling more than two classes, Regression, Unsupervised and descriptive learning. **Concept learning**: The hypothesis space, Paths through the hypothesis space, Beyond conjunctive concepts

UNIT- III: Tree models: Decision trees, Ranking and probability estimation trees, Tree learning as variance reduction. **Rule models:**Learning ordered rule lists, Learning unordered rule sets, Descriptive rule learning, First-order rule learning

UNIT -IV:Linear models: The least-squares method, The perceptron: a heuristic learning algorithm for linear classifiers, Support vector machines, obtaining probabilities from linear classifiers, Going beyond linearity with kernel methods.**Distance Based Models:** Introduction, Neighbours and exemplars, Nearest Neighbours classification, Distance Based Clustering, Hierarchical Clustering.

UNIT- V:Probabilistic models: The normal distribution and its geometric interpretations, Probabilistic models for categorical data, Discriminative learning by optimising conditional likelihoodProbabilistic models with hidden variables.**Features:** Kinds of feature, Feature transformations, Feature construction and selection. Model ensembles: Bagging and random forests, Boosting

UNIT- VI: Dimensionality Reduction: Principal Component Analysis (PCA), Implementation and demonstration. **Artificial Neural Networks:**Introduction, Neural network representation, appropriate problems for neural network learning, Multilayer networks and the back propagation algorithm.

OUTCOMES:
- Recognize the characteristics of machine learning that make it useful to real-world
- Problems.
- Characterize machine learning algorithms as supervised, semi-supervised, and
- Unsupervised.
- Have heard of a few machine learning toolboxes.
- Be able to use support vector machines.
- Be able to use regularized regression algorithms.
- Understand the concept behind neural networks for learning non-linear functions.

TEXT BOOKS:
1. Machine Learning: The art and science of algorithms that make sense of data, Peter Flach, Cambridge.
2. Machine Learning, Tom M. Mitchell, MGH.

REFERENCE BOOKS:
1. UnderstandingMachine Learning: From Theory toAlgorithms, Shai Shalev-Shwartz, Shai Ben-
David, Cambridge.
2. Machine Learning in Action, Peter Harington, 2012, Cengage.

	L	**T**	**P**	**C**
IV Year – II Semester	**4**	**0**	**0**	**3**

CONCURRENT AND PARALLEL PROGRAMMING
(Elective - 3)

OBJECTIVES:
- Improvement of students comprehension of CPP, new programming concepts, paradigms and idioms
- Change of 'mood' regarding Concurrency counter-intuitiveness
- Proactive attitude: theoretical teaching shouldn't be so dull
- Multipath, individually paced, stop–and–replay, personalized learning process
- Frequent assessment of learning advances on the subject

UNIT- 1
Concurrent versus sequential programming. Concurrent programming constructs and race condition. Synchronization primitives.

UNIT-II
Processes and threads. Interprocess communication. Livelock and deadlocks, starvation, and deadlock prevention. Issues and challenges in concurrent programming paradigm and current trends.

UNIT-III
Parallel algorithms – sorting, ranking, searching, traversals, prefix sum etc.,

UNIT- IV
Parallel programming paradigms – Data parallel, Task parallel, Shared memory and message passing, Parallel Architectures, GPGPU, pthreads, STM,

UNIT-V
OpenMP, OpenCL, Cilk++, Intel TBB, CUDA

UNIT-VI
Heterogeneous Computing: C++AMP, OpenCL

OUTCOMES:

- Understanding improvement of CPP concepts presented
- The number of reinforcement–exercises assigned
- The time required for the resolution of exercises
- Compliance level with the new model of theoretical teaching

TEXT BOOKS:

1. Mordechai Ben-Ari. Principles of Concurrent and Distributed Programming, Prentice-Hall International.
2. Greg Andrews. Concurrent Programming: Principles and Practice, Addison Wesley.
3. GadiTaubenfeld. Synchronization Algorithms and Concurrent Programming, Pearson.
4. M. Ben-Ari. Principles of Concurrent Programming, Prentice Hall.
5. Fred B. Schneider. On Concurrent Programming, Springer.
6. Brinch Hansen. The Origins of Concurrent Programming: From Semaphor

ARTIFICIAL NEURAL NETWORKS
(Elective-3)

OBJECTIVES:

- Understand the role of neural networks in engineering, artificial intelligence, and cognitive modeling.
- Provide knowledge of supervised learning in neural networks
- Provide knowledge of computation and dynamical systems using neural networks
- Provide knowledge of reinforcement learning using neural networks.
- Provide knowledge of unsupervised learning using neural networks.
- Provide hands-on experience in selected applications

UNIT-I: Introduction and ANN Structure.

Biological neurons and artificial neurons. Model of an ANN. Activation functions used in ANNs. Typical classes of network architectures.

UNIT-II

Mathematical Foundations and Learning mechanisms.Re-visiting vector and matrix algebra. State-space concepts. Concepts of optimization. Error-correction learning. Memory-based learning. Hebbian learning. Competitive learning.

UNIT-III

Single layer perceptrons. Structure and learning of perceptrons. Pattern classifier - introduction and Bayes' classifiers. Perceptron as a pattern classifier. Perceptron convergence. Limitations of a perceptrons.

UNIT-IV: Feed forward ANN.

Structures of Multi-layer feed forward networks. Back propagation algorithm. Back propagation - training and convergence. Functional approximation with back propagation. Practical and design issues of back propagation learning.

UNIT-V: Radial Basis Function Networks.

Pattern separability and interpolation. Regularization Theory. Regularization and RBF networks.RBF network design and training. Approximation properties of RBF.

UNIT-VI: Support Vector machines.

Linear separability and optimal hyperplane.Determination of optimal hyperplane. Optimal hyperplane for nonseparable patterns.Design of an SVM.Examples of SVM.

OUTCOMES:

- This course has been designed to offer as a graduate-level/ final year undergraduate level elective subject to the students of any branch of engineering/ science, having basic foundations of matrix algebra, calculus and preferably (not essential) with a basic knowledge of optimization.
- Students and researchers desirous of working on pattern recognition and classification, regression and interpolation from sparse observations; control and optimization are expected to find this course useful. The course covers theories and usage of artificial neural networks (ANN) for problems pertaining to classification (supervised/ unsupervised) and regression.
- The course starts with some mathematical foundations and the structures of artificial neurons, which mimics biological neurons in a grossly scaled down version. It offers mathematical basis of learning mechanisms through ANN. The course introduces perceptrons, discusses its capabilities and limitations as a pattern classifier and later develops concepts of multilayer perceptrons with back propagation learning.

TEXT BOOKS:

1. Simon Haykin, "Neural Networks: A comprehensive foundation", Second Edition, Pearson Education Asia.
2. Satish Kumar, "Neural Networks: A classroom approach", Tata McGraw Hill, 2004.

REFERENCE BOOKS:

1. Robert J. Schalkoff, "Artificial Neural Networks", McGraw-Hill International Editions, 1997.

OPERATION RESEARCH
(Elective - 3)

OBJECTIVE:
- Identify and develop operational research models from the verbal description of the real system.
- Understand the mathematical tools that are needed to solve optimisation problems.
- Use mathematical software to solve the proposed models.
- Develop a report that describes the model and the solving technique, analyse the results and propose recommendations in language understandable to the decision-making processes in Management Engineering

UNIT-I:
Introduction to Operations Research: Basics definition, scope, objectives, phases, models and limitations of Operations Research. Linear Programming Problem – Formulation of LPP, Graphical solution of LPP. Simplex Method, Artificial variables, big-M method, two-phase method, degeneracy and unbound solutions**.**

UNIT-II:
Transportation Problem. Formulation, solution, unbalanced Transportation problem. Finding basic feasible solutions – Northwest corner rule, least cost method and Vogel's approximation method. Optimality test: the stepping stone method and MODI method

UNIT-III:
Assignment model. Formulation. Hungarian method for optimal solution. Solving unbalanced problem. Traveling salesman problem and assignment problem Sequencing models. Solution of Sequencing Problem – Processing n Jobs through 2 Machines – Processing n Jobs through 3 Machines – Processing 2 Jobs through m machines – Processing n Jobs through m Machines

UNIT-IV:
Dynamic programming. Characteristics of dynamic programming. Dynamic programming approach for Priority Management employment smoothening, capital budgeting, Stage Coach/Shortest Path, cargo loading and Reliability problems Games Theory. Competitive games, rectangular game, saddle point, minimax (maximin) method of optimal strategies, value of the game. Solution of games with saddle points, dominance principle. Rectangular games without saddle point – mixed strategy for 2 X 2 games

UNIT-V:
Replacement Models. Replacement of Items that Deteriorate whose maintenance costs increase with time without change in the money value. Replacement of items that fail suddenly: individual replacement policy, group replacement policy

UNIT-VI:

Inventory models. Inventory costs. Models with deterministic demand – model (a) demand rate uniform and production rate infinite, model (b) demand rate non-uniform and production rate infinite, model (c) demand rate uniform and production rate finite.

OUTCOME:

- Methodology of Operations Research.
- Linear programming: solving methods, duality, and sensitivity analysis.
- Integer Programming.
- Network flows.
- Multi-criteria decision techniques.
- Decision making under uncertainty and risk.
- Game theory. Dynamic programming**.**

TEXT BOOKS:

1. P. Sankara Iyer,"Operations Research", Tata McGraw-Hill, 2008.
2. A.M. Natarajan, P. Balasubramani, A. Tamilarasi, "Operations Research", Pearson Education, 2005**.**

REFERENCE BOOKS:

1. J K Sharma. "Operations Research Theory & Applications, 3e", Macmillan India Ltd, 2007.
2. P. K. Gupta and D. S. Hira, "Operations Research", S. Chand & co., 2007.
3. J K Sharma., "Operations Research, Problems and Solutions, 3e", Macmillan India Ltd
4. N.V.S. Raju, "Operations Research", HI-TECH, 2002

IV Year – II Semester

L	T	P	C
0	3	0	2

SEMINAR

IV Year – II Semester

L	T	P	C
0	0	0	10

PROJECT