Academic Scheme

for

B. Tech. Computer Science and Engineering

(Batch-2018 and onwards)

Department of Computer Science and Engineering
Islamic University of Science and Technology
Awantipora, J&K, India-192122

Academic Scheme

for

B. Tech. Computer Science and Engineering

(Batch-2018 and onwards)

Approved in Board of Studies held on 20th December, 2018

Department of Computer Science and Engineering
Islamic University of Science and Technology
Awantipora, J&K, India-192122

Department of Computer Science and Engineering

The Department of Computer Science and Engineering was established in 2007 to provide technical skills and expertise in the fast-growing field of Computer Science and Engineering. The Department is well recognized for excellence in facilities and teaching. The Department offers 4 years B. Tech program in Computer Science and Engineering with an objective to enable students to acquire specialized knowledge for various subjects in Computer Science and Engineering. The department is also focused to enrich student's personal, social and cognitive development to meet the contemporary world challenges.

The department has highly qualified and experienced faculty, who are involved in interdisciplinary research work, which create tremendous opportunities for the students to carry out the novel and independent research in the field of Science and Engineering. The primary aim of the department is to create world class Engineering Leaders, who can contribute to the global community development and hence become the saviors in the current world emerging problems.

Board of Studies - 2018

The first Board of Studies (BoS) meeting of the Department of Computer Science and Engineering (CSE), Islamic University of Science and Technology (IUST) was held on 20th December 2018. It won't be out of place to mention that earlier five (05) BoS meetings for CSE were conducted jointly by School of Engineering and Technology (SoE&T). The following members attended the meeting:

1.	Prof. A.H. Moon	Dean SoE&T, IUST	Chairman
2.	Prof. Rohi Naaz	Dept. of CSE, NIT Srinagar	Academic Expert
3.	Mr. S. Yasser Kazmi	CEO, Myasa Network Solutions	Industrial Expert
4.	Mr. Mudasir Bhat	Lelafe IT Solution Pvt. Ltd	Industrial Expert
5.	Mr. Arshid Dar	Lelafe IT Solutions Pvt. Ltd	Industrial Expert
6.	Mr. Syed Mujtiba Hussain	I/C Head CSE	Convener
7.	Dr. Assif Assad	Assistant Professor, CSE	Member
8.	Mr. Sheikh Asif	DR Academics, IUST	Member

Additionally, all faculty members of CSE also attended the meeting.

Contents

Credit definition, range, distribution, and Course codes

A. Definition of Credit:

Credit is one of the primary methods used to determine and document that student has met academic requirements. Credits are awarded upon completing and passing a course.

1 hr. Lecture (L) per week	1 credit
1 hr. Tutorial (T) per week	1 credit
1 hr. Practical (P) per week	0.5 credit
2 hrs. Practical (P) per week	1 credit

B. Range of Credits:

Credits earned in the range of 163 and above shall be required for a student to be eligible to get Under Graduate Degree in Computer Science and Engineering. A student is eligible to get Under Graduate Degree with Honors or additional Minor Engineering, if he/she completes an additional 20 credits through MOOCs.

Those Students who are not interested in obtaining honors degree, may earn at least 2 additional credits through MOOCs so as to get familiar with online mode of learning.

C. Distribution of Credits

All students of B Tech., Department of Computer Science and Engineering have to **earn 8 credits** from **open elective** starting from **3rd semester and onwards** to complete the requirement of the Programme as per the Choice-Based Credit System (CBCS) implemented by Islamic university of Science and Technology (IUST), Awantipora.

S No.	***Semester***	***Total Credits***
01	*I*	*20*
02	*II*	*19*
03	*III*	*19*
04	*IV*	*20*
05	*V*	*21*
06	*VI*	*19*
07	*VII*	*18*
08	*VIII*	*19*
09	*Open Elective*	*08*
Total Credits		***163***

D. Course Code and Definition:

All courses (except Open Electives) are denoted by a seven-digit alphanumeric code (XXXXXXX), three alphabets followed by three numerals, followed by one alphabet.

1. The first three alphabets designate the department teaching the course, i.e., the discipline to which the course belongs, e.g., CSE for Computer Science & Engineering.

2. The first numeral following the three alphabets indicate the level of the course, 1 to 4 for undergraduate 1st to 4th year; 5 to 7 for postgraduate 1st to 3rd year, and 8, 9 for PhD.
3. The next two numerals are the unique identification numbers for the course. Courses running in odd semesters are labelled from 01 to 49 and courses running in even semesters are labelled from 50 to 99.
4. The last alphabet indicates the nature of the course. It is one amongst four choices, C (Core Course), E (Elective (Discipline Centric)), G (Elective (Generic)), F (Foundation Course). Since, open electives are identified by a zero in place of the level numeral (at fourth digit), therefore the last digit does not have significance in their course code, and hence will not be used for the definition of the same.
5. Open Electives have a zero in place of the above level numeral and thereby six digits only.

Course Structure for B.Tech. Batch 2018 onwards

Semester-I

Course Code	Course Title	Hours Per Week		Credits
		L	P	
PHY101C	Physics	4	0	4
CHM101C	Chemistry	4	0	4
MTH103C	Mathematics-I	3	0	3
BIO101F	Environmental Science	3	0	3
MEC101C	Engineering Graphics and Design	1	4	3
ENG101F	Communication Skills	2	2	3
-	Induction Programme	-	-	-
Total Credits		**17**	**6**	**20**

Semester-II

Course Code	Course Title	Hours Per Week		Credits
		L	P	
CIV150C	Engineering Mechanics	**3**	**0**	**3**
MTH153C	Mathematics –II	**4**	**0**	**4**
CSE150F	Programming for Problem Solving	**3**	**0**	**3**
MEC150C	Workshop Practices	**1**	**4**	**3**
ELE150C	Basic Electrical Engineering	**3**	**0**	**3**
CSE151F	Programming Lab	**0**	**2**	**1**
PHY150C	Physics Lab	**0**	**2**	**1**
CHM150C	Chemistry Lab	**0**	**2**	**1**
Total Credits		**14**	**10**	**19**

Semester-III

Course Code	Course Title	Hours Per Week		Credit
		L	P	
CSE201C	Object Oriented Programming	3	0	3
CSE202C	Software Engineering	3	0	3
CSE203C	Digital Electronics and Logic Design	3	0	3
CSE204C	Discrete Mathematics	3	0	3
ECE231C	Basic Analog Electronics	3	0	3
CSE205C	Object Oriented Programming Lab	0	4	2
CSE206C	Digital Electronics and Logic Design Lab	0	2	1
ECE232C	Basic Analog Electronics Lab	0	2	1
Total Credits		**15**	**8**	**19**

Semester-IV

Course Code	Course Title	Hours Per Week		Credit
		L	P	
CSE250C	Data Structures	3	0	3
CSE251C	Computer Architecture and Organization	3	0	3
CSE252C	Data Communication	3	0	3
CSE253C	Internet and Web Designing	1	4	3
CSE254C	Linux Internals	2	2	3
STA271C	Probability & Statistics	3	0	3
CSE255C	Data Structures Lab	0	4	2
Total Credits		**15**	**10**	**20**

Semester-V

Course Code	Course Title	Hours Per Week		Credit
		L	P	
CSE301C	Design and Analysis of Algorithms	3	0	3
CSE302C	Operating System	3	0	3
CSE303C	Database Management System	3	0	3
CSE304C	Computer Networks	3	0	3
CSE305C	Numerical Methods	3	0	3
CSE306C	Simulation Lab	1	4	3
CSE307C	Design and Analysis of Algorithms Lab	0	2	1
CSE308C	Database Management System Lab	0	2	1
CSE309C	Computer Networks Lab	0	2	1
Total Credits		**16**	**10**	**21**

Semester-VI

Course Code	Course Title	Hours Per Week		Credit
		L	P	
CSE350C	Microprocessors, Peripherals and Interfacing	3	0	3
CSE351C	Formal Language and Automata Theory	3	0	3
CSE352C	Artificial Intelligence	3	0	3
CSE353C	Java Programming	3	0	3
	Elective- I	3	0	3
CSE354C	Microprocessors, Peripherals and Interfacing Lab	0	2	1
CSE355C	Java Programming Lab	0	2	1
CSE356C	Project I	0	2	1
CSE357C	Industrial Training	0	0	1
Total Credits		**15**	**6**	**19**

Semester-VII

Course Code	Course Title	Hours Per Week		Credit
		L	P	
CSE401C	Compiler Design	3	0	3
CSE402C	Network Security	3	0	3
	Elective –II	3	0	3
	Elective –III	3	0	3
CSE403C	Compiler Design Lab	0	2	1
CSE404C	Network Security Lab	0	2	1
CSE405C	Project II	0	6	3
CSE406C	Seminar	0	0	1
Total Credits		**12**	**10**	**18**

Semester-VIII

Course Code	Course Title	Hours Per Week		Credit
		L	P	
DMS471C	Organisational Behaviour	3	0	3
	Elective –IV	3	0	3
	Elective –V	3	0	3
CSE450C	Project–III	0	20	10
Total Credits		**9**	**20**	**19**

List of Discipline Centric Electives

Course Code	Course Title	Hours Per Week		Credits	Pre-requisite	Semester
		L	P			
CSE350E	Computer Graphics	3	0	03	Computer Programming	6th
CSE351E	Introduction to Core Java Script	2	2	03	Internet and Web Designing	6th
CSE352E	Software Testing and Quality Assurance	3	0	03	Software Engineering	6th
CSE353E	C# and .Net Programming	2	2	03	Computer Programming	6th
CSE354E	Distributed Computing	3	0	03	OS	6th
CSE355E	Digital Image Processing	3	0	03	Computer Programming	6th
CSE356E	Programming in in R	2	2	03	Nill	6th
CSE357E	Soft Computing	3	0	03	Computer Programming	6th
CSE401E	Advanced JavaScript	2	2	03	Introduction to Core Java Script	7th
CSE402E	Python Programmin	2	2	03	Nill	7h
CSE403E	Machine Learning	3	0	03	Computer Programming	7th
CSE404E	Multimedia Technology	3	0	03	Nill	7th
CSE405E	Web Technologies	2	2	03	I&WD and JAVA	7th
CSE406E	Advanced Java	2	2	03	Java	7th
CSE407E	Data Mining	3	0	03	DBMS	7th
CSE408E	Robotics	3	0	03	Embedded system	7th
CSE409E	Internet of Things	3	0	03	Computer Networks/ Embedded Systems	7th
CSE410E	Agile Software Development	3	0	03	Software Engineering	7th
CSE411E	GO Language	2	2	03	Nill	7th
CSE412E	Modelling and Simulation	3	0	03	Computer Programming	7th

List of Discipline Centric Electives

CSE450E	Big Data	3	0	03	DBMS	8th
CSE451E	Advanced Computer Architecture	3	0	03	CAO	8th
CSE452E	High Performance Computing	3	0	03	Computer Architecture & Organization	8th
CSE453E	Computer Vision	3	0	03	DS	8th
CSE454E	Operation Research	3	0	03	Nill	8th
CSE455E	Deep Learning	3	0	03	Machine Learning/DIP	8th
CSE456E	Ethical Hacking	3	0	03	Computer Network	8th
CSE457E	Cloud Computing	3	0	03	Computer Networks	8th
CSE458E	Natural language processing	3	0	03	Computer Programming	8th
CSE359E	Mobile Application Development	3	0	03	Java	8th
CSE460E	Wireless Communication	3	0	03	Computer Networks/Data Communication	8th
CSE461E	Reinforcement Learning	3	0	03	Machine Learning	8th
CSE462E	Evolutionary Computing	3	0	03	Computer Programming	8th
CSE463E	Software Project Management	3	0	03	Software Engineering	8th
CSE464E	Computational Ethics Design	3	0	03		8th
CSE465E	Bioinformatics	3	0	03	Computer Programming	8th
CSE-466E	Block Chain	3	0	03	Computer Network, Network Security	8th

List of Open Electives

Course Code	Course Title	Hours Per Week		Credits
		L	P	
CSE001	Introduction to Linux	1	2	02
CSE002	Introduction to Matlab	1	2	02
CSE003	Introduction to Internet and Web Application	1	2	02
CSE004	Basic Computer Applications	1	2	02
CSE005	Computer Fundamentals	1	2	02
CSE006	Latex	1	2	02

B-Tech Computer Science & Engineering
Syllabus for Semester-I

Course Code	Course Title	Hours Per Week		Credits	Page No.
		L	P		
PHY101C	Physics	4	0	4	12
CHM101C	Chemistry	4	0	4	13-14
MTH103C	Mathematics-I	3	0	3	15
BIO101F	Environmental Science	3	0	3	16
MEC101C	Engineering Graphics and Design	1	4	3	17
ENG101F	Communication Skills	2	2	3	18
-	Induction Programme	-	-	-	
Total Credits		**17**	**6**	**20**	

Course Code: **PHY101C**	Course Title: **Physics**	Credits: 04 **L - 4 P – 0**

Course Outcomes

- *Learn vector calculus approach to solve the problems of electric field and magnetic field.*
- *Learn and apply profound understanding of Quantum Mechanics and its applications for problem solving.*
- *Understand special theory of relativity and its concepts in detail.*
- *Use the relevant parameters required to realize various physical phenomenon in devices using lasers, holograms, antireflection coating, refractive index, thickness etc.*
- *Understand the wave optics concepts such as interference, diffraction and their applications.*
- *Learn about nuclear physics, nuclear fission and fusion, radioactivity etc.*

Vectors: Vector Analysis, Rotation of coordinate axis and Transformation of vectors, Gradient of scalar field, divergence and curl of vector field in Cartesians, Spherical polar and Cylindrical Coordinate systems, line, surface & volume integrals, Gauss's divergence theorem, Stokes's theorem.

Mechanics: Newton's laws of motion**,** rigid body, centre of mass, conservation of linear momentum, moment of inertia, conservation of angular momentum, Central forces, Keplers laws for planetary motion.
SHM, Damped, undamped and forced Oscillations (no derivation): Equation of motion, solution, amplitude resonance, velocity resonance, quality factor.

Special theory of Relativity: Frame of reference, Michelson-Morley experiment, Galilian transformations, basic postulates of special relativity, Lorentz transformations, length contraction and time dilation, mass energy relation.

Optics: Electromagnetic theory of light, Interference**:** Conditions for Interference of light, Young's double slit experiment, Newton's rings, diffraction**:** Single Slit diffraction pattern, Diffraction grating, Grating spectra, Polarization: Malus Law, Phenomena of double refraction.

Lasers: Properties of laser light**,** Main components of laser, absorption, spontaneous and stimulated emission, CW and pulsed lasers, Examples and applications: He-Ne laser, Ruby laser.

Quantum Theory: Need of Quantum theory, Photoelectric effect, Compton Effect, Heisenberg's uncertainty principle, de Broglie's hypothesis. Basic postulates of quantum mechanics, Wave function and its properties, Schrodinger's equation and its application to particle in 1-D box.

Nuclear physics: Structure of nucleus. Basic properties of nucleus (size, charge, and density), Binding energy, nuclear fission & fusion, Radioactivity, Gas detectors: GM counter.

Elementary Solid-State Physics: Crystal lattice, Crystal structure, Unit cells, Miller Indices, Bravais lattice, Bragg's Law, Photographic crystal X-ray diffraction techniques, Laue's method. Free electron theory of metals, Classification of solids, formation of energy bands in metals, semiconductors and insulators, intrinsic and extrinsic semiconductors.

Text Books/Reference Books:

1. Griffiths D. J., Introduction to electrodynamics, *Pearson Education (India).*
2. Murray R. Speigel, Schaum's Outline on Vector Analysis, *McGraw Hill Education India.*
3. Upadhaya J. C., Classical Mechanics, *Himalaya Publishing House.*
4. Ghatak A., Optics, *McGraw Hill Education India.*
5. Besier A., Mahajan S., Choudhary S. R., Concepts of Modern Physics, *McGraw Hill Education India.*
6. Omar M. A., Elementary Solid-State Physics, *Prentice Hall of India*.

Course Code: **CHM101C**	Course Title: **Chemistry**	Credits: 04 **L - 4 P – 0**

Course Outcomes

- *Understand the importance of thermodynamics, laws of thermodynamics along with some other important concepts such as entropy, Carnot engine, Gibbs free energy etc.*
- *Gain the basic knowledge of electrochemical procedures related to corrosion and its control.*
- *Understand about different polymers and nano materials.*
- *Understand the basic concept of lubricants and their applications in various fields of Engineering.*
- *Have a detailed knowledge of instrumental and non-instrumental methods.*

Chemical Thermodynamics: Introduction and Importance, First Law of Thermodynamics, Work done in Isothermal and Adiabatic Conditions, Heat capacities, Relation between C_p and C_v relations, Second Law of Thermodynamics, Concept of Entropy, Carnot engine, Gibbs free energy. Free Energy Changes as Criteria of Reversible and Irreversible process, Gibbs-Helmholtz's equation, Clausius–Clapeyron equation

Electro-Chemistry and Corrosion: Introduction, Conductivity of Electrolytes, Kohlrausch's Law of Independent Migration of Ions and its Application, Debye Huckel Theory of Strong Electrolytes. Electrochemical cells, Electrode-Potential, Standard Electrode Potential, Fuel Cells, Batteries, Introduction, Effects of Corrosion, Dry Corrosion and Wet Corrosion, mechanisms, Types of Corrosion (Pitting Corrosion, Crevice Corrosion, Galvanic Corrosion and Stress corrosion), Factors Effecting Corrosion (Nature of the Metal and Nature of the Environment), Corrosion Protection and Inhibition (Cathodic Protection, Anodic Protection, Protective Coatings)

Nano-Technology and Polymers: Nanoscale and Its Significance, Properties at Nanoscale: Optical, Electrical, and Magnetic. General Methods of Preparation of Nanomaterials viz Top Down (Ball Milling, Lithography) and Bottom up Methods (Sol-Gel, Solution Based Method), Advantages of Polymers over other Engineering Materials, Functionality, Degree of Polymerization, Concept of Molecular Weight, Polymerization (Addition, Condensation and Copolymerization), Polymerization Techniques (Bulk, Solution, Suspension and Emulsion polymerizations), Preparation, Properties and Engineering application of some Important Polymers, Polythene (LDPE and HDPE), Polyvinyl Chloride, Polystyrene, Teflon, Phenol Formaldehyde, urea-formaldehyde resin

Lubricants: Introduction, Function of Lubricants, Mechanism of Lubrication, Classification of Lubricants (Liquid, Semisolid, Solid), Properties of Lubricants (Flash Point and Fire Point, Viscosity, Aniline Point Acid value)

Instrumental Techniques: Introduction, Advantages and Disadvantages of Instrumental and Non-Instrumental Methods, Electromagnetic Radiation, Electromagnetic Spectrum, Light Absorption (Beers-Lambert Law) UV-Vis spectroscopy (Types of Transition, Chromophors, Auxo-chromes and Applications), Infrared Spectroscopy (Modes of vibration, IR bands corresponding to different functional groups and Applications), Nuclear Magnetic Resonance: Principle, shielding mechanism, chemical shift, number of Signals, Application of Nuclear Magnetic Resonance to Simple Organic Molecules.

Text Books/Reference Books:

1. Chemistry in Engineering and Technology Volumes I & II, J. Kuriacose, R. Rajaram, 2001, *TMH publishing company Limited, New Delhi.*
2. Engineering Chemistry, P.C. Jain, 16th Edition, *Dhanpat Rai & Sons, Nai Sarak; New Delhi*.
3. Chemistry of Engineering Materials, C.V. Agarwal, 9th Edition.
4. Chemistry in Engineering, L. A. Munro, 1964, *Prentice Hall, New York.*
5. Applied Chemistry for Engineers, R. M. E. Diamant, 3rd Revised Edition, *Pitman Publishing.*
6. Principles of Physical Chemistry – Puri, Sharma and Pathania, 2017, 4th Edition, *Vishal Publishing Co.*
7. Physical Chemistry by Peter Atkins, Julio de Paula, 8th Edition, 2006, *WH Freeman.*
8. Concise Inorganic Chemistry by J.D. Lee, 5th Edition, 2008, *Oxford University Press.*
9. Electrochemistry and Corrosion Science by N. Perez, 2nd Edition, 2016, *Springer.*
10. Polymer Science, V.R. Goowriker, N.V Viswanathan and Jayadev Sreedhar, 2nd Edition, 2015, N*ew Age International Publishers.*
11. Nanotechnology Fundamentals and Applications, Manasi Karkare, Rajni Bahuguna, 2013, *I K international.*
12. Nanotechnology Importance and Application, Fulekar, 2010, K *International Publishing House.*

Course Code: **MTH103C**	Course Title: **Mathematics-I**	Credits: 03 **L - 3 P – 0**

Course Outcomes

- *Understand and explain the concepts of differential calculus.*
- *Learn about ordinary differential equations, second and higher order differential equations and their applications.*
- *Explain the concept of Vector Differentiation*
- *Apply the matrix techniques to reduce the quadratic forms to canonical forms, finding solutions of systems of linear equations in the different areas of Linear Algebra.*
- *Learn about fundamentals of algebra and algebraic equations.*

Brief Review of Differential Calculus: Limit, continuity and differentiability of functions of several variables, Chain rule, Jacobi theorem. Taylor's theorem of one and two variables, extrema of functions, two or more variables using method of Lagrange's multipliers.

Ordinary Differential Equations: Exact ordinary differential equations and Ordinary differential equations reducible to exact differential equations. Linear differential equations and equations reducible to linear form. Linear Differential equations of second and higher order with constant and variable coefficients. Applications of ordinary differential equations. Series solution of differential equations.

Vector Spaces: Linear dependence of vectors, Basis and Dimensions; Linear Transformations (maps), Range and Kernel of a linear map, Rank and Nullity, Inverse of a linear transformation, Rank-Nullity Theorem, Composition of Linear maps, Matrix associated with a linear map.

Algebraic Equations, Elements of the theory of polynomial equations. Fundamental theorem of Algebra, Relation between the roots and the coefficients of an equation, Solution of cubic & bi-quadratic equations.

Text Books:

1. Shanti Narayan, Differential calculus, *S. Chand & Sons*.
2. J. W. Brown, R. V. Churchill, Complex variables and Applications, *McGraw Hill Education India.*
3. Raisinghania M. D., Ordinary and Partial Differential equation, *S. Chand & Sons*.
4. Kreyszig I., Advanced Engineering Mathematics, *John Wiley & Sons.*

Reference Books:

1. James Stewart, Calculus, *Early Transcedentals*.
2. Bali N. P., A text Book on Engineering Mathematics, *Luxmi Publications.*
3. Jain R. K., Iyengar S. R. K., Advanced Engineering Mathematics, *Narosa Publications.*
4. Hoffmann & Kunze, Linear Algebra, *Prentice Hall of India.*
5. Piaggio H. T., Differential equations and its applications, *Prentice Hall of India*.
6. Sastry, Engineering mathematics Vol I-II, *Prentice Hall of India*.

Course Code: **BIO101F**	Course Title: **Environmental Science**	Credits: 03 **L - 3 P – 0**

Course Outcomes

- *Have an awareness about public environment methods, environment issues and ethics.*
- *Understand the principles of ecology and environmental issues that apply to air, land, and water issues on a global scale.*
- *Have a knowledge of natural resources, biodiversity and threats to biodiversity.*
- *Learn about Global Climate Change and various other concepts including Green House effect, Global Warming, Acid rain and Ozone layer depletion.*
- *Visit some local sites to study about some natural environmental resources.*

Introduction to Environmental Science: Scope and importance, Public Environmental awareness and methods of its propagation, Consumerism and Green Consumerism. Environmental issues, Environmental Ethics-Anthropocentricism and Ecocentricism.

Introduction to Ecosystem and Ecology: Types of Ecosystems, Structure of an Eco system-biotic and abiotic components, Food chain and Food Web, Ecological Pyramids; Ecological Succession, Energy flow in an ecosystem, Major World Ecosystems and their characteristics.

Natural resources: Classification and their conservation; Biodiversity-Definition, values and threats to biodiversity; Classification of species as per IUCN; Hot Spots of Biodiversity. Conservation approaches – *In-Situ* and *Ex-Situ* conservation; Alternatives to conventional developmental approaches – Sustainable Development.

Introduction to global climate change: Greenhouse effect, global warming, acid rain, ozone layer depletion. Definition, Cause, effects and control measures of Air pollution, water pollution, soil pollution, noise pollution, thermal pollution and Solid waste pollution.

Field work (Field work equal to 5 lecture hours), Visit to a local area to document environmental assets river/forest/grassland/hill/mountain. Visit to a local polluted site-Urban/Rural/Industrial/Agricultural. Study of common plants, insects, birds. Study of simple ecosystems-pond, river, hill slopes, etc.

Text Books/Reference Books:

1. Ecology and Environment, P. D. Sharma, *Rastogi Publications*.
2. Environmental Science Towards a Sustainable Future, Nebel and Wright, *Prentice Hall of India.*
3. Environmental Studies, Erach Barucha, *Oxford Publications.*
4. Environmental Studies from Crises to Cure authored, R. Rajagopalan, *Oxford University Press.*
5. Environmental Management by Oberoi, *Excel Books.*
6. Principles of Environmental Science: Inquiry & Applications, William Cunningham & Mary Cunningham, *Tata McGraw Hill.*
7. Perspectives of environmental studies, A. P. Kaushik and C.P. Kaushik, *New Age International Publications.*

Course Code: **MEC101C**	Course Title: **Engineering Graphics and Design**	Credits: 03 **L - 1 P – 4**

Course Outcomes

- *Use the drawing instruments and be able to dimension the given figures*
- *Prepare multi view orthographic projections of objects by visualizing them in different positions*
- *Understand the concept of projection and acquire visualization skills, projection of points*
- *Draw sectional views and develop surfaces of a given object*
- *Able to draw the basic views related to projections of Lines, Planes*
- *Prepare pictorial drawings using the principles of isometric and perspective projections to visualize objects in three dimensions.*
- *Obtain multi view projections and solid models of objects using CAD tools*

Introduction: Principles of Engineering Graphics; Orthographic Projection; Descriptive Geometry; Drawing Principles; Isometric Projection; Surface Development; Perspective; Reading a Drawing; Sectional Views; Dimensioning & Tolerances; True Length, Angle; intersection, Shortest Distance, Drawing instruments, lettering, Conic sections; Cycloid, Epicycloid, Hypocycloid and Involute; Scales.

Orthographic Projections: Principles of Orthographic Projections, Conventions, Projections of Points and lines inclined to both planes; Projections of planes inclined Planes, Auxiliary Planes;

Projections of Solids: Auxiliary Views; Draw simple annotation, dimensioning and scaling. Floor plans that include: windows, doors, and fixtures such as WC, bath, sink, shower, etc.

Sections of Solids: Prism, Cylinder, Pyramid, Cone, Auxiliary Views; Development of surfaces; sectional orthographic views, objects from industry and dwellings.

Isometric Projections: Principles of Isometric projection, Isometric Scale, Isometric Views, Conventions; Isometric Views of lines, Planes, Simple and compound Solids; Conversion of Isometric Views to Orthographic Views and Vice-versa

Overview of Computer Graphics: Computer technologies, CAD software, the Menu System, Toolbars, Standard, Object Properties, Draw, Modify and Dimension, Drawing Area (Background, Crosshairs, Coordinate System), Dialog boxes and windows, Shortcut menus, Different commands used in CAD, Isometric Views of lines, Planes, Simple and compound Solids.

Customisation & CAD Drawing: Set up of the drawing page and the printer, including scale settings, setting up of units and drawing limits; ISO and ANSI standards for coordinate dimensioning and tolerancing; Orthographic constraints.

Text Books:

1. Gill P. S., Engineering Drawing, *S. K. Kataria and sons.*
2. Bhatt N. D., Engineering Drawing, *Charotar Book Stall.*
3. James D. Bethune, Engineering Graphics with Auto CADD, *Pearson Education.*

Reference Books:

1. Shah M. B., Rana B. C., Engineering Drawing and Computer Graphics, *Pearson Education.*
2. Agrawal B., Agrawal C. M., Engineering Graphics, *TMH Publication.*

Course Code: **ENG101F**	Course Title: **Communication Skills**	Credits: 03 **L - 2 P – 2**

Course Outcomes

- *To get acquainted with concept of word formation, foreign languages and other methods of vocabulary building.*
- *To learn about organizing principles of paragraphs in documents, punctuation, sentence structures etc. to enhance writing skills.*
- *To identify and remove redundancies, clichés and common errors in writing.*
- *Will learn the skills for sensible writing.*
- *To get acquainted with proper pronunciation, methods of conducting formal presentations, interviews to enhance oral communication skills.*

Vocabulary Building: The concept of Word Formation, Root words from foreign languages and their use in English, Acquaintance with prefixes and suffixes from foreign languages in English to form derivatives, Synonyms, antonyms, and standard abbreviations.

Basic Writing Skills, Sentence Structures, use of phrases and clauses in sentences, Importance of proper punctuation, creating coherence, organizing principles of paragraphs in documents, Techniques for writing precisely.

Identifying Common Errors in Writing: Subject-verb agreement, Noun-pronoun agreement, Misplaced modifiers, Articles, Prepositions, Redundancies, Clichés.

Nature and Style of sensible Writing: Describing, Defining, Classifying, providing examples or evidence, Writing introduction and conclusion

Writing Practices: Comprehension, Précis Writing, Essay Writing.

Oral Communication: (This unit involves interactive practice sessions in Language Lab): Listening Comprehension, Pronunciation, Intonation, Stress and Rhythm, Common Everyday Situations: Conversations and Dialogues, Communication at Workplace, Interviews, Formal Presentations

Text Books/Reference Books:

1. Michael Swan, Practical English Usage, *OUP, 1995*.
2. Wood F. T., Remedial English Grammar, *Macmillan, 2007*.
3. William Zinsser, On Writing Well, *Harper Resource Book, 2001*.
4. Liz Hamp-Lyons and Ben Heasly, Study Writing, *Cambridge University Press, 2006*.
5. Sanjay Kumar and Pushp Lata, Communication Skills, *Oxford University Press, 2011*.
6. Exercises in Spoken English. Parts. I-III. CIEFL, Hyderabad, *Oxford University Press.*

B-Tech Computer Science & Engineering

Syllabus for Semester-II

Course Code	Course Title	Hours Per Week		Credits
		L	P	
CIV150C	Engineering Mechanics	**3**	**0**	**3**
MTH153C	Mathematics –II	**4**	**0**	**4**
CSE150F	Programming for Problem Solving	**3**	**0**	**3**
MEC150C	Workshop Practices	**1**	**4**	**3**
ELE150C	Basic Electrical Engineering	**3**	**0**	**3**
CSE151F	Programming Lab	**0**	**2**	**1**
PHY150C	Physics Lab	**0**	**2**	**1**
CHM150C	Chemistry Lab	**0**	**2**	**1**
	Total Credits	**14**	**10**	**19**

Course Code: **CIV150C**	Course Title: **Engineering Mechanics**	Credits: 03 **L - 2 P – 4**

Course Outcomes

- *To make the students to know the importance of Mechanics, equation of static equilibrium and dynamic equilibrium of particles and rigid bodies.*
- *Determine the centroid and moment of area of sections.*
- *To learn effect of friction on equilibrium.*
- *Determine Center of gravity and mass movement of inertia of solid objects.*
- *To learn Kinematics of particles and rigid body, related principles.*

Force Systems: Basic concepts, equilibrium of rigid bodies, system of forces, coplanar concurrent forces, components in space, resultants, moment of forces and its application, couples and resultant of force system, equilibrium of system of forces, free body diagrams, equations of equilibrium of coplanar systems and spatial systems, static indeterminacy.

Centroid and Second Moment of Area: Centroid of simple figures from first principle, centroid of composite sections; Area moment of Inertia**,** Moment of Inertia of plane sections from first principles, theorems of moment of inertia, moment of inertia of standard sections and composite sections.

Basic Structural Analysis: Equilibrium of deformable bodies, external and internal forces, stresses and strains in bars, basic introduction to beams, shear force and bending moment in simple beams, basic introduction to torsion, and analysis of trusses using method of joints.

Friction: Types of friction, limiting friction, dry friction, laws of friction, static and dynamic friction; motion of bodies, wedge friction, screw jack, friction clutches and brakes.

Centre of Gravity and Moment of Inertia: Centre of gravity and its implications; Mass moment of inertia, Moment of inertia of Cylinder, Cone, Sphere, etc.

Fundamentals of Dynamics: Kinematics and Kinetics of particles in rectilinear and curvilinear motion; Kinematics and Kinetics of Rigid bodies, types of motion,

instantaneous centre of rotation in plane motion, D'Alembert's principle and its applications in plane motion and connected bodies, Work Energy principle, Impulse-Momentum principle, Impact.

Text Books:

1. Irving H. Shames, Engineering Mechanics, *Prentice Hall India, New Delhi.*
2. R. C. Hibbler, Engineering Mechanics: Principles of Statics and Dynamics, *Pearson Education.*

Reference Books:

1. F. P. Beer, E. R. Johnston, Vector Mechanics for Engineers, Vol I & Vol II, *McGraw Hill Education (India).*
2. Andy Ruina and Rudra Pratap, Introduction to Statics and Dynamics, *Oxford University Press.*
3. Shanes and Rao, Engineering Mechanics, *Pearson Education.*
4. Hibler and Gupta, Engineering Mechanics (Statics, Dynamics), *Pearson Education.*
5. Bansal R. K., A Text Book of Engineering Mechanics, *Laxmi Publications.*

Course Code: **MTH153C**	Course Title: **Mathematics-II**	Credits: 04 **L - 4 P – 0**

Course Outcomes

- *Able to apply integral calculus in different fields of engineering*
- *Able to understand the uses of differential equations to solve various problems*
- *Able to interpret and analyze data with the help of linear algebra*
- *After studying complex variables students will be able to find solutions for different complex analysis problems*
- *Able to apply analytic functions to represent and analyze given data*

Integral Calculus: Definite Integrals and their properties, Differential under the sign of integration. Double and triple integrals, Change of variables, Beta and Gamma functions, Fourier series.

Non-linear differential equation of first order, Simultaneous differential equation, Simultaneous differential equation of the form dx/P = dy/Q = dz/R . Partial differential equations of first order, langrage linear equation, Standard form, Charpit`s Method to solve non- linear partial differential equation, Partial differential equations of second and higher order, Homogeneous Partial Differential equations with constant coefficients, Solutions by the method of separation of variables, heat flow equation, Wave equation.

Matrices: Eigen values and Eigen vectors of a matrix, Cayley-Hamilton Theorem, Symmetric, Skew-symmetric, Hermitian, skew- Hermitian, Orthogonal and unitary matrices and their properties, Diagonalization; Inner product spaces, Gram-Schmidt Orthogonalization.

Complex Variables: Differentiation, Cauchy-Riemann Equations, Analytic functions, Harmonic functions, elementary analytic functions(exponential, logarithmic and trigonometric) and their properties, Taylor's series and Laurent's series.

Text Books:

1. Kreyszig I., Advanced Engineering Mathematics, *John Wiley & Sons.*
2. Piaggio H. T., Differential equations and its applications, H *Prentice Hall of India*.
3. Raisinghania M. D., Ordinary and Partial Differential equation, *S. Chand & Sons*.

Reference Books:

1. James Stewart, Calculus, *Early Transcedentals*.
2. Hoffmann & Kunze, Linear Algebra, *Prentice Hall of India.*
3. Shanti Narayan, Integral Calculus by Shanty Narayan, *S. Chand & Sons*.
4. Greenberg, Advanced Engineering Mathematics, *Pearson education.*
5. Sastry, Engineering mathematics Vol I-II, *Prentice Hall of India*.

Course Code: **CSE150F**	Course Title: **Programming for Problem Solving**	Credits: 03 **L - 3 P – 0**

Course Outcomes

- *Introduction to problem solving using C programming.*
- *Design and implement algorithms for simple problems.*
- *Define, understand and describe C programming concepts like data types, control Structures, Pointers, Dynamic Memory Allocation.*
- *Understanding modular approach for problem solving using functions*
- *Introduction to File handling using C programming*

Introduction to Programming: Introduction to components of a computer system (disks, memory, processor, where a program is stored and executed, operating system, compilers etc. Idea of Algorithm: steps to solve logical and numerical problems. Representation of Algorithm: Flowchart/Pseudocode with examples. From algorithms to programs; source code, variables (with data types) variables and memory locations, Syntax and Logical Errors in compilation, object and executable code.

Branching, Loops, and Arrays: Arithmetic expressions and precedence, Conditional Branching and Loops, Writing and evaluation of conditionals and consequent branching, Iteration and loops. Arrays, Arrays (1-D, 2-D), Character arrays and Strings.

Algorithms, Order complexity and Functions: Basic Algorithms, Searching, Basic Sorting Algorithms (Bubble, Insertion and Selection), Finding roots of equations, notion of order of complexity through example programs (no formal definition required), Function, Functions (including using built in libraries), Parameter passing in functions, call by value, Passing
arrays to functions: idea of call by reference.

Recursion: Recursion as a different way of solving problems. Example programs, such as Finding Factorial,
Fibonacci series, Ackerman function etc. Quick sort or Merge sort, Structure, Structures, Defining structures and Array of Structures.

Pointers: Idea of pointers, Defining pointers, Use of Pointers in self-referential structures, notion of linked list (no implementation), File handling.

Text Books:

1. E. Balaguruswamy, Programming in ANSI C, *McGraw Hill Education India.*
2. Yashavant Kanetkar, Let Us C, *BPB Publications*

Reference Books:

1. Gottfried, Schaum's Outline of Programming with C, *McGraw Hill Education India.*
2. Brian W. Kernighan and Dennis M. Ritchie, The C Programming Language, *Prentice Hall of India.*

Course Code: **MEC150C**	Course Title: **Workshop Practices**	Credits: 03 **L - 1 P – 4**

Course Outcomes

- *Able to understand the principle of metal cutting and preparation of jobs on different machines*
- *Able to prepare a wooden model using fitting and carpentry shop*
- *Able to understand various power tools to perform various operations*
- *Able to understand the principle and working of joining using different welding techniques*
- *Able to understand the principle and working of mould preparation and casting using smithy shop*

(i) Lectures and Videos

1. Manufacturing Methods- casting, forming, machining, joining, advanced manufacturing methods.
2. CNC machining, Additive manufacturing.
3. Fitting operations & power tools.
4. Electrical & Electronics.
5. Carpentry.
6. Plastic moulding, glass cutting.
7. Metal casting.
8. Welding (arc welding & gas welding), brazing.

(ii) Workshop Practice

1. Machine shop
2. Fitting shop
3. Carpentry
4. Electrical & Electronics
5. Welding shop (Arc welding, gas welding)
6. Casting
7. Smithy Shop

Text Books:

1. Hajra Choudhury S.K., Hajra Choudhury A.K. and Nirjhar Roy S.K., Elements of Workshop Technology, Vol. I 2008 and Vol. II 2010, *Media promoters and publishers private limited, Mumbai.*
2. Kalpakjian S. And Steven S. Schmid, Manufacturing Engineering and Technology, 4th edition, *Pearson Education India Edition, 2002.*

Reference Books:

1. Gowri P. Hariharan and A. Suresh Babu, Manufacturing Technology – I, *Pearson Education, 2008.*
2. Roy A. Lindberg, Processes and Materials of Manufacture, 4th edition, *Prentice Hall India, 1998.*
3. Rao P.N., Manufacturing Technology, Vol. I and Vol. II, *Tata McGraw Hill House, 2017.*

Course Code: **ELE150C**	Course Title: **Basic Electrical Engineering**	Credits: 03 **L - 3 P – 0**

Course Outcomes

- *To get acquainted with the detailed knowledge of electrical elements (R, L, C) and laws of electricity including KCL, KVL.*
- *To get acquainted with basic terminologies of AC and to make the students aware about the importance of power factor with phasor diagrams.*
- *To discuss in detail the three-phase systems.*
- *To get acquainted with the basic knowledge of magnetic materials and their properties and to study the construction, principle and working of transformers.*
- *To get the students acquainted with detailed knowledge of electrical machines.*
- *To get the students acquainted about the basic procedure of electrical installations including components, fuses, MCB, SWG, batteries and earthing.*

DC Circuits: Electrical circuit elements (R, L and C), voltage and current sources, Kirchoff current and voltage laws, analysis of simple circuits with dc excitation. Superposition, Thevenin and Norton Theorems.

AC Circuits: Representation of sinusoidal waveforms, peak and rms values, phasor representation, real power, reactive power, apparent power, power factor. Analysis of single-phase ac circuits, resonance in series and parallel RLC circuits. Three phase balanced circuits, voltage and current relations in star and delta connections.

Transformers: Magnetic materials, BH characteristics, ideal and practical transformer, equivalent circuit, losses in transformers, regulation and efficiency. Three-phase transformer connections.

Electrical Machines: Generation of rotating magnetic fields. Construction and working of a three-phase induction Motor. Significance of torque-slip characteristic. Starting of induction motor. Construction, working, torque-speed characteristic of separately excited dc motor. Construction and working of synchronous generators.

Electrical Installations: Components of LT Switchgear: Switch Fuse Unit (SFU), MCB, ELCB, MCCB, Types of Wires and Cables, Earthing. Types of Batteries, Important Characteristics for Batteries.

Text Books:

1. D. P. Kothari and I. J. Nagrath, Basic Electrical Engineering, *Tata McGraw Hill, 2010.*
2. D. C. Kulshreshtha, Basic Electrical Engineering, *McGraw Hill, 2009.*
3. V. D. Toro, Electrical Engineering Fundamentals, *Prentice Hall India, 1989.*

Reference Books:

1. E. Hughes, Electrical and Electronics Technology, *Pearson, 2010.*
2. Charles K. Alexender, Mathew N. O. Sadiku, Fundamentals of Electric circuits, *McGraw Hill,*
3. Jack E. Kemmerly William H. Hayt, Engineering Circuit Analysis, *McGraw Hill, 2012.*
4. L. S. Bobrow, Fundamentals of Electrical Engineering, *Oxford University Press, 2011.*

Course Code: **CSE151F**	Course Title: **Programming Lab**	Credits: 01 **L - 0 P – 2**

Course Outcomes

- *Introduction to problem solving using C programming.*
- *Design and implement algorithms for simple problems.*
- *Define, understand and describe C programming concepts like data types, control Structures, Pointers, Dynamic Memory Allocation.*
- *Understanding modular approach for problem solving using functions*
- *Introduction to File handling using C programming*

List of Experiments

1. Familiarization with the programming environment
2. Simple computational problems using arithmetic expressions
3. Problems involving if-then-else structures
4. Iterative problems e.g., sum of series
5. 1D Array manipulation
6. Matrix problems, String operations
7. Simple functions
8. Programming for solving Numerical methods problems
9. Recursive functions
10. Pointers and structures
11. File operations

Course Code: **PHY150C**	Course Title: **Physics Lab**	Credits: 01 **L - 0 P – 2**

Course Outcomes

- *Able to find out optical properties of different light sources, and velocity of sound wave*
- *Able to determine the acceleration due to gravity by measuring time period of different pendulum*
- *Able to study V-I characteristics of different electronics devices*
- *Able to study properties of semiconductors using Hall effect*
- *Able to verify different physical laws*

List of Experiments

1. To determine the value of e/m of an Electron by Thompson Method
2. To determine the value of Acceleration due to gravity(g) by using Bar Pendulum
3. To determine the value of Acceleration due to gravity(g) by using Kater's Reversible Pendulum
4. To determine the Young's Modulus of rigidity of rectangular Steel Bar by Bending of Beam Method.
5. To determine the Wavelength of Sodium Light by Newton's Rings.
6. To determine the Wavelength of Laser Source by Fresnel Biprism
7. To determine the frequency of AC by Melde's Method
8. To determine The Resolving Power of Telescope.
9. To study the moment of Inertia of a Fly Wheel
10. To determine the refractive index of Crown Glass Prism.
11. To determine the wavelength of Sodium Light by Plane diffraction Grating.
12. To study the characteristics of Zener Diode.
13. To determine the Wavelength of Prominent lines of Mercury Light by Plane Diffraction Grating.
14. To study the characteristics of PN Junction Diode (Forward Bias)
15. To verify Biot-Savart's Law by showing that magnetic field produced is directly proportional to the current passed in a coil.
16. To study the characteristics of G.M. Tube.
17. To determine Planck's constant by LED Method.
18. To verify Stefan's Law by Electrical method.
19. Determination of Modulus of rigidity by Maxwell's Needle
20. Determination of velocity of Sound by Standing Wave Method.
21. To study the Hall Effect:
    (i) Determination of Hall Voltage and RH.
    (ii) Determination of mobility of charge carriers and carrier concentration

Course Code: **CHM150C**	Course Title: **Chemistry Lab**	Credits: 01 **L - 0 P – 2**

Course Outcomes

- *Able to prepare solutions in various concentrations, handling Glass wares, and fine chemicals with precautions in the chemical laboratory*
- *Determine strength of bases using acid-base titrimetric analysis*
- *Able to find Aniline point of lubricant oil, moisture content of coal and ion exchange capacity of commercial resin*
- *Able to analyse hardness of water, alkalinity of water by Complexometric titration and disassociation constant of weak acid by PH meter*
- *Synthesis formaldehyde resin*

List of Experiments

1. Basic Introduction on Solution Preparation, Concentration terms, Handling of Glass ware, Chemicals, Instruments: Precautions.
2. Determination of strength of NaOH solution by standardization of sodium hydroxide using Oxalic acid
3. To determine the acid value of a given mineral oil or vegetable oil.
4. To determine the moisture content of a given sample of coal.
5. To determine the Degree of dissociation of a weak acid by Conductometry.
6. Determination of the strength and pK_a value of the weak acid by titration with an alkali.
7. To determine the Aniline point of the given sample of a Lubricating oil.
8. Synthesis of the phenol formaldehyde resin.
9. To determine the temporary and permanent hardness of a sample of water by complexometric titration.
10. To determine the Alkalinity of the given sample of water.
11. Determination of the ion exchange capacity of cation exchange resin.

Demonstration Experiments

1. Determination of pH of different concentration of acid and bases by pH meter.
2. Spectrophotometer (concentration determination, wavelength maximum)

Text Books/Reference Books:

1. Laboratory Manual on Engineering Chemistry by S. K. Bhasin, S. Rani, 2009, *D R Publications.*
2. J. B. Yadav, Advanced Practical Physical Chemistry.

B-Tech Computer Science & Engineering
Syllabus for Semester-III

Course Code	Course Title	Hours Per Week		Credit
		L	P	
CSE201C	Object Oriented Programming	3	0	3
CSE202C	Software Engineering	3	0	3
CSE203C	Digital Electronics and Logic Design	3	0	3
CSE204C	Discrete Mathematics	3	0	3
ECE231C	Basic Analog Electronics	3	0	3
CSE205C	Object Oriented Programming Lab	0	4	2
CSE206C	Digital Electronics and Logic Design Lab	0	2	1
ECE232C	Basic Analog Electronics Lab	0	2	1
Total Credits		**15**	**8**	**19**

Course Code: **CSE-311T**	Course Title: **Object Oriented Programming**	Credits: 04 **L - 4 P – 0**

Course Outcomes

- *Understanding the features of C++ supporting object-oriented programming.*
- *Understanding functions, classes, data and objects*
- *Understanding various types of constructors and destructor.*
- *Understanding operator overloading, virtual functions, polymorphism and inheritance with the understanding of early and late binding.*
- *Understanding advanced features of C++ specifically strings and templates.*

UNIT - I

Introduction: Basic features & concepts of Object-Oriented Programming (OOP), Benefits, Languages and Applications of OOPs.

Tokens, Expressions and Control Structures: Tokens, Keywords, Identifiers & Constants, Basic Data types, User-defined Data types, Derived Data Types, Memory Management Operators, Manipulators, Expressions, Operator Overloading, Control Structures.

UNIT - II

Functions in C++: Main function, function prototyping, call by reference, Returning more values by Reference, inline functions, default functions, function overloading. Static data members, static function members. **Classes and Objects:** Specifying a class, defining member functions, private member functions, array within a class, memory allocation for objects, arrays of objects, objects as function arguments, returning objects, pointers to members, local classes, Nested Class, this pointer.

UNIT - III

Constructors & Destructors: Constructors, Parameterized Constructors, Constructors with Default arguments, Dynamic Initialization of objects, Dynamic Constructors & Destructors, Recursive Constructor, Constructor and Destructor with Static Members. **Operator Overloading & Type Conversion:** Definition & Rules of overloading Operators, Overloading Binary & Unary Operators. Overloading operators using friends.

UNIT - IV

Inheritance: Definition, single, multilevel, multiple, hierarchical and hybrid inheritance, virtual base classes, abstract classes. **Pointers, Virtual Functions and Polymorphism:** Pointers, Pointers to Objects and derived classes, virtual functions, Pure virtual functions.

UNIT - V

Templates: Introduction, Function templates, Class templates.

Exception Handling: Introduction, Principles of Exception Handling, The Keywords Try, Throw and Catch, Exception Handling Mechanism, Multiple Catch Statements, Catching Multiple Exceptions, Re- Throwing Exception, Specifying Exception, Exceptions in Constructor and Destructors, Controlling Uncaught Exceptions

Text Books recommended:

1. *Object Oriented Programming with C++*, E Balagurusamy
2. *Object Oriented Programming in Turbo C++*, Robert Lafore
3. *Teach Yourself C++*, Al Stevens
4. *A Structured Approach using C++*, Farouzan & Gilberg
5. *Object Oriented Programming with C++*, R S Salaria.

Course Code: **CSE 202C**	Course Title: **Software Engineering**	Credits: 03 **L - 3 P - 0**

- *Understanding and applying the software engineering lifecycle models by demonstrating competence in communication, planning, analysis, design, construction, and deployment*
- *Translate a requirements specification into an implementable design, following a structured and organized process.*
- *Defining the basic concepts and importance of Software project planning concepts like cost estimation, and scheduling.*
- *Applying different testing and debugging techniques and analyzing their effectiveness.*
- *Defining software maintenance concepts, software quality and reliability on the basis of international quality standards.*

UNIT - I

Introduction: software engineering discipline-evolution and Impact, program vs software product, emergence of software Engineering. Software Crisis, Software Processes, Software life cycle models: Waterfall, Prototype, Evolutionary and Spiral models, Overview of Quality Standards like ISO 9001, SEI-CMM.

UNIT - II

Software Project Planning, project manager responsibilities, Cost estimation, static, Single and multivariate models, COCOMO model, Risk management. Software Requirement Analysis and Specifications: Problem Analysis, Data Flow Diagrams, Data Dictionaries, Entity-Relationship diagrams, Software Requirement and Specifications, Risk management, Software Configuration Management.

UNIT - III

Software Design: Cohesion & Coupling, Classification of Cohesiveness & Coupling, Function Oriented Design, Object Oriented Design, User Interface Design. Software Reliability: Failure and Faults, Reliability Models: Basic Model, Logarithmic Poisson Model, Calendar time Component, Reliability Allocation.

UNIT - IV

Software Testing: Software process, Functional testing: Boundary value analysis, Equivalence class testing, Decision table testing, Cause effect graphing, Structural testing: Path testing, Data flow and mutation testing, unit testing, integration and system testing, Black box testing, White- box testing, Debugging, Testing Tools & Standards. An introduction to software reliability theory

UNIT - V

Software Quality, Software Quality Management System ISO 9000, SEI CMM, Personal Software Process, Six Sigma, Software Maintenance, Reverse Engineering, Software Re-engineering, Configuration Management, Documentation.

Text Books:

1. "*Fundamental of Software Engineering",* by Rajeev Mall, PHI.
2. "*Software Engineering",* by James F. Peters, Wiley.
3. "*Software Engineering A. Practitioner's Approach*", by Pressman, MGH.

Reference Books:

1. *Software Project Management from Concept to Development,* by Kieron Conway, Dreamtech Press.
2. "*Software Engineering*", by Sommerville, Pearson Education.
3. "*Software Engineering",* by Jawadekar, TMH.

Course Code: **CSE-203C**	Course Title: **Digital Electronics and Logical design**	Credits: 03 **L - 3 P - 0**

- *To have deep understanding of digital logic levels, so to have a complete understanding between the efficiency of digital vs analog.*
- *To have a complete knowledge of number systems and their conversions and how to apply them in digital circuits and their applications in logic circuit design*
- *To have a knowledge of combinational circuits with basic functions like adder, subtractor, encoder, decoder, Multiplexer, de-multiplexer, comparators etc. To know the sequential circuits like flip-flops, counters etc. and their circuit implementation.*

UNIT-I

Number Systems and Codes

Introduction, Number systems and Codes: excess-3 code, gray code, error detection and correction codes. Boolean algebra and Reduction Techniques: binary logic, Postulates and theorems, logic functions, minimization of Boolean functions using algebraic, Karnaugh map and Quine – McClausky methods, Logic gates.

UNIT-II

Digital Logic circuits

Introduction to combinational circuit: Realization of basic combinational functions like Adders, BCD Adder, binary adder, Subtractors, Encoder - decoder, Multiplexer - Demultiplexer, Comparators, Delays and hazards in combinational circuits.

UNIT-III

Sequential Circuits

Flip-Flops: SR, JK, T, D, Master-Slave FF, Triggering of FF, Analysis of clocked sequential circuits - their design, State minimization, state assignment.

UNIT-IV

Registers and Counters

Registers: Shift registers, Counters: Asynchronous counters, Synchronous counters.

UNIT-V

Digital Logic Families and Data Converters

RTL, DTL, TTL, ECL, ICL, HTL, NMOS & CMOS logic gates, Circuit diagram, characteristics and specifications, tri-state gates. Analog to Digital converters, Digital to Analog converters, Programmable Logic Devices. (PLD's)

Text Books Recommended:

1. Morris Mano, "*Digital logic and Computer Design*", Prentice Hall of India.
2. Ronald J. Tocci, "*Digital Systems, Principles and Applications",* Prentice-Hall of India.
3. Jain R.P., "*Modern Digital Electronics",* Tata McGraw Hill.
4. Floyd T.L., "*Digital Fundamentals*", Charles E. Merrill Publishing Company.

Course Code: **CSE-204C**	Course Title: **Discrete Mathematics**	Credits: 03 **L - 3 P - 0**

Course Outcomes:

- *Understand the notion of mathematical thinking, mathematical proofs, and algorithmic thinking, and be able to apply them in problem solving.*
- *Understand the basics of discrete probability and number theory, and be able to apply the methods from these subjects in problem solving.*
- *To have ability of Counting and Solving Recurrence Relations.*
- *Modelling real world problems using discrete mathematics.*

UNIT - I

Finite Sets, Power Set, Cardinality of finite sets, Cartesian Product, Properties of Sets

UNIT - II

Introduction to Logic, Propositional Logic, Truth tables, Deduction, Resolution, Predicates and Quantifiers, Mathematical Proofs. Infinite sets, well-ordering. Countable and Uncountable sets, Mathematical Induction - weak and strong induction.

UNIT - III

Relations, Equivalence Relations. Functions, Bijections. Binary relations, Posets and Lattices, Hasse Diagrams.

UNIT - IV

Graphs and Multi-graphs, Degree of a vertex, Paths connectivity, Cut points Bridges, Walks, paths, cycles, connected graphs, Bipartite, Regular, Planar and connected graphs,Components, Euler graphs, Euler's theorem, Hamiltonian path and circuits, Graph colouring, chromatic number, isomorphism and Homomorphism of graphs, Konigsberg seven bridge problem, Shortest path. Trees, properties of trees, pendant vertices in trees, Degree sequences in trees, Necessary and sufficient conditions for a sequence to be a degree sequence of a tree.

UNIT - V

Counting, Sum and product rule, Principle of Inclusion Exclusion. Pigeonhole Principle, Boolean Algebra, Postulates of Boolean Algebra; Theorems of Boolean Algebra, Sum of products and product of sums Simplification, NAND and NOR implementation.

Books Recommended

1. C. L. Liu: *Elements of Discrete Mathematics*, 2nd Ed.Tata Mc-Graw Hill.
2. *Kolman, Busby and Ross: Discrete Mathematical Structures*, 6th Ed.PHI (2009).
3. Narsingh Deo: *Graph Theory with Applications to Engineering and Computer Sciences*, PHI.
4. Murry R. Spiegel**:** *Discrete Mathematics (Schaums Outline series)*, Tata McGraw Hill**,** (2009).

Reference Books

- Kenneth H. Rosen: *Discrete Mathematics and its applications*,5th Ed. Tata McGraw Hill (2003).
- K.R Parthasarty: *Basic Graph Theory*, Tata Mc-Graw Hill

Course Code: **ECE231C**	Course Title: **Basic Analog Electronics**	Credits: 03 **L - 3 P - 0**

Course Outcomes

- *Understand the working of pn-junction diodes and the various circuits made up of diodes.*
- *Understand the working of bi-polar transistors and their characteristics along with their applications.*
- *Understand the working of MOS FETs and contrast them with the BJT.*
- *Understand the elements of a communication system and the need for modulation.*
- *Understand the working of pn-junction diodes and the various circuits made up of diodes.*

UNIT - I

PN Junction: Review of crystal structure of semiconductors, Intrinsic and Extrinsic semiconductors, Basic structure of pn junction under zero, forward and reverse bias, Basic Diode Circuits, Circuit models, graphical and iterative analysis, load line analysis, rectifier circuits, voltage regulation, limiting circuits, level shifters, Zener diode

UNIT - II

Bipolar Transistor: Basic structure and principle of operation, modes of operation, static IV characteristics in active and saturation modes, amplification, minority carrier distribution, emitter efficiency, transport factor, current gain, non-ideal effects, small signal model and frequency limitations

UNIT - III

MOS Field Effect Transistor: MOS structure, energy band diagrams, static IV characteristics, analysis of MOS structure and C-V characteristics, short channel effects, small signal model and frequency limitations

UNIT - IV

Communication Systems: Introduction to Communication System, Elements of Communication System, Benefits of Communication, Communication Media, Modulation and Demodulation (brief idea)

Text/Reference Books:

1. Sedra A.S. and Smith K.C., *Microelectronic Circuits*, Oxford University Press.
2. Razavi B., *Fundamentals of Microelectronics*, John Wiley & Sons.
3. Boylestad R. and Nashelsky L., *Electronic Devices and Circuits*, Prentice Hall
4. Neamen D. A., *Microelectronics: Circuit Analysis and Design*, McGraw Hill Publications

Course Code: **CSE-250C**	Course Title: **Object Oriented Programming Lab**	Credits: 02 **L - 0 P – 4**

- *Apply C++ features to program design and implementation.*
- *Explain object-oriented concepts and describe how they are supported by C++.*
- *Use C++ to demonstrate practical experience in developing object-oriented solutions*
- *Analyze a problem description and design and build object-oriented software using good coding practices and techniques*
- *Use common software patterns in object-oriented design and recognize their applicability to other software development contexts.*

Lab Details

1. Function overloading, default arguments in C++.static member and static functions

2. Simple class design in C++, namespaces, objects creations, Array of objects

3. Class design in C++ using dynamic memory allocation.

4. Destructor, copy constructor, constructor overloading

5. Operator overloading, friend functions

6. Overloading assignment operator, type conversions

7. Inheritance, run-time polymorphism

Course Code: **CSE206C**	Course Title: **Digital Electronics and Logical Design Lab**	Credits: 01 **L - 0 P – 2**

- *How digital values of analog signals are represented in different logic families.*
- *Description of truth tables for digital logic circuits.*
- *Gate level implementation of No. of circuits like multiplexers etc.*
- *Evaluation of No. of constraints (parameters) of combinational and sequential logic circuits.*
- *To implement the logic circuits in the design of VLSI / IC circuits using hardware descriptive languages like VHDL, VERILOG etc.*

Lab Details

1. To verify the truth table of the following logic gates:
   - AND, OR, NOT
   - NAND, NOR, XOR, X-NOR

2. Realization of:
   - Half Adder and verify its truth table
   - Full Adder and verify its truth table
   - Half subtractor and verify its truth table
   - Full subtractor and verify its truth table
   -

3. To design multiplexer and demultiplexer using 2-input NAND gates.

4. Realization of:
   - Encoders and Decoders
   - Flip-Flops (SR, JK, T, D flip-flops)

Course Code: **ECE232C**	Course Title: **Basic Analog Electronics Lab**	Credits: 01 **L - 0 P – 2**

- *How digital values of analog signals are represented in different logic families.*
- *Description of truth tables for digital logic circuits.*
- *Gate level implementation of No. of circuits like multiplexers etc.*
- *Evaluation of No. of constraints (parameters) of combinational and sequential logic circuits.*
- *To implement the logic circuits in the design of VLSI / IC circuits using hardware descriptive languages like VHDL, VERILOG etc.*

List of Experiments

1. Familiarity with lab equipment like multimeter, ammeter, voltmeter, bred board, CRO, power supplies, etc.
2. Familiarity with electrical/electronic components like resistors, inductors, capacitors, diodes, LEDs, etc.
3. Color coding of resistors
4. Series and parallel combination of resistors.
5. Verification of Ohms Law, Kirchoff's Laws.
6. Voltage divider and current dividers.
7. Verification of Superposition, Thevenin and Norton Theorems.
8. Hands on soldering and desoldering techniques
9. Steady State Characteristics of pn junction under different bias conditions
10. Static IV characteristics of bipolar transistors
11. Static IV characteristics of MOSFETs

B-Tech Computer Science & Engineering
Syllabus for Semester-IV

Course Code	Course Title	Hours Per Week		Credit
		L	P	
CSE250C	Data Structures	3	0	3
CSE251C	Computer Architecture and Organization	3	0	3
CSE252C	Data Communication	3	0	3
CSE253C	Internet and Web Designing	1	4	3
CSE254C	Linux Internals	2	2	3
STA271C	Probability & Statistics	3	0	3
CSE255C	Data Structures Lab	0	4	2
Total Credits		**15**	**10**	**20**

Course Code: **CSE-250C**	Course Title: **Data Structures**	Credits: 03 **L - 3 P - 0**

Course Outcomes:

- *Define, understand, describe and implement linear data structures like stack, queues, linked lists and non-liner data structures like trees and graphs.*
- *Design and trace the algorithms for various operations on different data structures studied.*
- *Understand and Implement various searching and sorting techniques*
- *Write programs in C to simulate operations and applications of data structures learnt*

Unit I:

Introduction to data structures, classification of Data Structures, Primitive vs. Non-Primitive data structures, Linear vs Non-Linear data structures, Primitive Data Structures Operations, Recursion Function & its Examples. String Manipulation.

Unit II:

Linked Lists -Singly, Doubly and Circular, their Implementation & Comparison, Concept of Stack & Queue, Array Based & Linked List Based Implementation of Stack & Queue & their Applications. Different types of queues.

Unit III:

Searching: Sequential & Binary Search on Array-based Ordered Lists, Binary Trees, their Implementation & Traversal, Binary Search Trees: Searching, Insertion & Deletion of Nodes, Height Balanced Trees & Concept of AVL Trees, Concept & purpose of B Trees & B+ Trees.

Unit IV:

Graphs: Definition, Terminology & Representation using Adjacency Matrix & Linked List. Graph Traversals: BFS & DFS algorithms & their Implementations. Spanning Tree.

Unit V:

Sorting Techniques: Insertion Sort, Selection Sort, Merge Sort, Quick Sort, Heap Sort, Shell Sort, Radix Sort. Concept of Hash Functions, Hash-tables & Hashing with Chaining.
File Structure: Sequential Files, Indexed Files, Direct Files.

Books Recommended:

1. Shaum's outlines "*Data Structures with C*" Seymour Lipschutz, Tata McGraw Hill Education.
2. Langsam Augenstein Tenenbaum "*Data Structures using C and C++"*

Reference Books:

- Tremblay & Sorenson, "*An Introduction to Data Structures with Applications*", McGraw hill, Kongakusha.
- Horowitz Sahni Mehta, "*Fundamentals of Data structures*", SBCS Publication.

Course Code: **CSE-251C**	Course Title: ***Computer Architecture and Organization***	Credits: 03 **L - 3 P - 0**

Course Outcomes:

- *Understand how the different functional units in computer system operate, interact and communicate.*
- *Understand the concrete representation and computation of data at the machine level.*
- *Understand how the I/O devices are being accessed and its principle*

UNIT - I

Register Transfer and Micro-operations: What is Computer Architecture, what is Computer Organization, Register Transfer Language, Register Transfer, Bus and Memory Transfers, Arithmetic Micro-operations, Logic Micro-operations, Shift Micro-operations, Arithmetic Logic Shift Unit.

UNIT - II

Basic Computer Organization and Design: Control Organization – Hardwired and Microprogrammed control. Instruction Codes, Computer Registers, Computer Instructions, Timing and Control, Instruction Cycle, Memory Reference Instructions, Input-Output and Interrupt, Complete Computer Description, Design of Basic Computer, Control Memory, Address Sequencing, Micro program Example, Design of Control Unit.

UNIT - III

Central Processing Unit: Introduction, General Register Organization, Stack Organization, Instruction Formats, Addressing Modes, Data Transfer and Manipulation, Program Control.
Computer Arithmetic: Addition and Subtraction, Multiplication Algorithms, Decimal Arithmetic Unit.

UNIT - IV

Input-Output Organization: Peripheral Devices, Input-Output Interface, Asynchronous Data Transfer, Modes of Transfer, Priority Interrupt, Direct Memory Access (DMA).
Pipelining: Pipelining Basics, Arithmetic Pipeline, Instruction Pipeline.

UNIT - V

Memory Organization: Memory Hierarchy, Main Memory, Auxiliary Memory, Associative Memory, Cache Memory, Virtual Memory.

Text Book: Morris Mano, "*Computer System Architecture",* Third Edition, PHI.

Reference Books:

1. W. Stallings, "*Computer Organization & Architecture*", PHI.
2. J. P. Hayes, "*Computer Architecture and Organization*", McGraw Hill.
3. J. L Hennessy and D. A. Patterson, "*Computer Architecture: A quantitative approach*", Morgon Kauffman, 1992.
4. "*Computer Systems Organization and Architecture*", John D. Carpinelli, Pearson Education Inc.

Course Code: **CSE252C**	Course Title: **Data Communication**	Credits: 03 **L - 3 P - 0**

Course Outcomes:

- *Introduce students to the digital and analogue representations of signals and channels and mathematical foundation of communication.*
- *Understand the working and implementation of various digital modulation schemes.*
- *Understand how various media work and the scenarios where they fit best.*
- *Understand the essence of multiplexing and its various techniques.*
- *Understand the working of various network technologies.*

Unit I

Modulation & Need for modulation, Amplitude modulation, AM modulation index, spectrum of AM signal, power analysis of AM signal, Frequency modulation, Frequency modulation index, Narrow band and broad band FM signal. Capture effect, Comparison between FM and AM, SNR of FM.

Unit II

Data and Signals: Data, Signals, Types of Signals, Bandwidth, spectrum, Digitization of analog signals, sampling, Nyquist sampling theorem, quantization, quantization noise, Pulse code modulation

Digital Modulation Techniques: ASK, FSK, PSK, DPSK, M-ary PSK, QAM. Signal constellation. Line coding techniques: NRZ, RZ, Biphase, Manchester coding, AMI, HDBn

Unit III

Transmission media: Guided and un-guided media, twisted wire pair, coaxial cable, optical fibre, microwave links, satellite microwave link, their characteristic features and applications for data transmission.

Data transmission: simplex, half duplex and full duplex, Asynchronous and synchronous data transmission. Carrier, bit and frame synchronization techniques, Phase lock loop.

Unit IV

Multiplexing Techniques: Frequency Division Multiplexing, Time Division Multiplexing, Wavelength division Multiplexing and Code Division Multiplexing. Spread Spectrum.

Errors in data communication: Types of errors, error detection and correction techniques, forward error correction, polynomial error detection scheme, computation of CRC.

Unit -V

Data communication network: Basic concept of network, Advantages and applications, Types of networks (LAN, MAN and WAN), Different network topologies like star, ring, hybrid, tree. Introduction to OSI model, TC-ip.

Books recommended

1. William Stallings: Data & Computer Communications, 9th Ed, PHI
2. Data Communications and Networking: Behrouz A. Forouzan
3. Andrew Tanenbaum, “Computer Networks” PHI
4. Sklar, “Digital Communications fundamentals & Applications” 2nd Ed Pearson Pub.
5. Keizer, “ Local Area Networks” McGraw Hill

Course Code: **CSE-253C**	Course Title: **Internet and Web Designing**	Credits: 03 **L - 1 P - 4**

Course Outcomes:

- *Able to design and format web pages using HTML and CSS.*
- *By learning java script able to handle events and client-side processing*
- *Create web sites using graphics and media objects in Web page*
- *Gain familiarity with the principles of electronic documentation and structured documents, particularly the XML standard family.*
- *Overall learning of Web Designing will improve students coding and designing concepts*

Unit – I: Introduction to Web Development.

Why Web is Important. Briefly how internet works (client – server Architecture). Understanding HTTP Request types ('GET', 'POST', 'PUT', 'DELETE') and Response status codes. Introduction to Front-end, Back-end and Full Stack Development. Distinguish and Differentiate between them. Introduction to fundamentals of a web page elements HTML, CSS and JS. How these work together for a complete web experience. Why web is the future of application development. Current trends in web development like Progressive Web Apps, electronJS.

Unit – II: Introduction to HTML

Explain how HTML is structure of a web page. Different HTML standards and present HTML standard. Difference and contrast between them. Dissecting a simple HTML document, understanding the essence of Head and Body of a HTML document. Basic HTML tags: paragraph tag, anchor tag, heading tags, image tag, lists (unordered, ordered, definition) tags. Various attributes related to tags. Tables in HTML, how information is represented in the form of tables. Forms, input tags like checkboxes, drop-down, radio box, text area, button and more. Building a simple HTML page using all these tags. Inline and Block Elements within the context of HTML. Using iframes. Logically related elements, divisions and spans.

Unit – III: HTML 5 the newest standard

Semantic nature of HTML 5. Introduction to Semantic HTML elements, need for semantic elements. Using HTML 5 elements such as navigation, section, main, aside, article, header, footer for defining the structure of a web page. HTML 5 audio video tags. Form validation using HTML 5 build-in attributes. Demonstrate your knowledge about HTML 5 by creating a website that enforces semantic nature of HTML 5. Brief introduction to HTML 5 API especially Web Storage.

Unit – IV: Introduction to CSS

Need for CSS. How to include styles in HTML, inline, internal and external stylesheets. Targeting specific elements, Children, Descendant and Attribute Selectors. Concept of Classes and IDs. Pseudo Classes. The Box-Model. Positioning Elements, relative, absolute, sticky, fixed. Using Floats and clearing floats. Understanding the default stacking order of elements and z-index. Units absolute and relatives, difference between the two. Understanding how CSS is parsed by the Browser and Inheritance in regard to CSS. Different phases of parsing CSS. Introduction to Responsive Design. Implementing Responsive Design using view-port, media query, and relative units. Create a website which is optimized for both large screens and phones using media queries and fluid layout using CSS 3. Exploration of new CSS 3 features like transform, animations etc.

Unit – V: Introduction to JavaScript

Understanding why JavaScript is the most popular language. Why to Master JS, Future of JS. Data Types, Array and Methods, Conditionals, using let and const instead of var keyword. Document Object Model. Accessing and mutating the DOM using JS. Custom form validations using JS. Create a basic custom form validator.

Text Books:

1. HTML and CSS: Design and Build Websites – Jon Duckett
2. JavaScript and JQuery: Interactive Front–End Web Development – Jon Duckett

Reference Books:

3. Greenlaw R and Hepp E "*Fundamentals of Internet and www*".
4. B. Underdahle and K.Underdahle, "*Internet and Web Page / Website Design",* DG Books India (P) Ltd.
5. D. Comer, "*The Internet Book*", Prentice Hall of India.

Course Code: **CSE254C**	Course Title: **Linux Internals**	Credits: 03 **L - 2 P - 2**

Course Outcomes:

- *Defining Operating system concepts and its types. Understanding different Linux distributions and installing Linux in virtual and real environment.*
- *Understanding the basic set of commands and utilities in Linux/UNIX systems and Linux firewall. Managing packages and users in Linux environment.*
- *Setting up internet services such as DNS, FTP and public key cryptographic protocols in Linux.*
- *Understanding, Installing and administering file systems and other services such as NFS, SAMBA, DHCP.*
- *Demonstrating Linux process control and shell programming.*

Unit - I:

Introduction, history of Linux, Linux terminology, Linux distributions, services associated with distributions, boot process, master boot record and boot loader, BIOS, text mode login.

Unit - II:

The Linux kernel, /sbin /init and services, system features, Linux file systems, partitions, file system hierarchy standards, Linux installation: the process.

Unit - III:

Graphical desktop, X-window system, GNOME desktop environment, logging in and out in Ubuntu without using GUI, switching users in Ubuntu, common applications.

Unit IV:

Command line operations, basic operations, working with files, searching for files, installing software's, introduction to text editors, finding Linux documentation, Linux processes, process metrics and process control.

Unit V:

File systems, file system architecture, comparing files and file types, bash shell and basic scripting, syntax: build in shell commands, script parameters, environment variables, functions.

Text Books Recommended:

1. Wale Soyinka, "*Linux administration: A beginners guide*", 5th edition, Tata McGraw Hill.
2. Steven Graham, Steve Shah, "*Linux administration*", 3rd edition, Dreamtech press.
3. Richard Peterson, "*Linux: the complete reference*", 6th edition, Tata McGraw Hill.

Online resources:

1. *www.linux.org*.
2. *www.linux.com*.
3. *www.linuxhomenetworking.com*.
4. *https://www.linuxfoundation.org*.

Course Code: **STA271C**	Course Title: **Probability and Statistics**	Credits: 03 **L - 3 P - 0**

Course Outcomes:

- *Solve basic probability problems*
- *Understand the random variables and various related concepts like joint, conditional and marginal probabilities.*
- *Identify and work with different distributions*
- *Appreciate the common ground between probability and statistics.*
- *Solve basic probability problems*

Unit-I:

Statistics: Measures of central tendency and Measures of variations (Dispersions), Moments, Measures of Skewness and Kurtosis. Moment generating functions, problems.

Unit-II:

Probability: Random experiment, sample space, events, classical, statistical and axiomatic definitions of probability. Statements and proof of theorems on addition and multiplication of probabilities.

Unit-III:

Conditional Probability: Bayes theorem on conditional probability. Random variables, Derivation of formulae for mean, variance and moments of random variables for discrete and continuous cases. Laws of expectation.

Unit-IV:

Standard Distributions: Bernoulli and Binomial distributions, Poisson and Normal Distributions, Beta and Gamma Distribution, t-Distribution, F-Distribution, Chi-square Distribution. Central Limit Theorem.

Unit-V:

Method of Least Squares & Correlation: Methods of least squares, fitting of straight line and parabola of degree 'p'. Regression and Correlation. Multiple and Partial Correlation. Problems

Books Recommended:

1. *Fundamentals of Mathematical Statistics*, by S.C.Gupta and V.K. Kapoor, Sulltan Chand & Sons New Delhi, Latest edition.
2. *Statistical Theory and Methodology in Science & Engineering*, by Brownlee, John Wiley & Sons.
3. *Introduction to Mathematical Statistics*, by R.E.Walpole 3rd edition New York Macmillan publication.
4. *Data Analysis for Scientists & Engineers*, by Meyer, John Wiley & Sons.

Course Code: **CSE255C**	Course Title: **Data Structures Lab**	Credits: 02 **L - 0 P – 4**

Course Outcomes:

- *Define, understand, describe and implement linear data structures like stack, queues, linked lists and non-liner data structures like trees and graphs.*
- *Design and trace the algorithms for various operations on different data structures studied.*
- *Understand and Implement various searching and sorting techniques*
- *Write programs in C to simulate operations and applications of data structures learnt*

Lab Details:

1. Program on arrays.
2. Implementation of String Manipulation
3. Programs on Recursion.
4. Implementation of single and doubly linked list and various operations on them.
5. Applications of Linked List, Polynomial Manipulation.
6. Circular Linked List.
7. Implementation of Stack and Queue using Array and linked list.
8. Applications of stack like Tower of Hanoi, Conversion of Infix Expression to polish notation, Conversion of Polish Expression to Code.
9. Implementation of Double Ended Queue
10. BST and its implementation.
11. Implementation of Sorting techniques- Selection Sort, Insertion Sort, Merge Sort, Quick Sort, Radix Sort, Heap Sort, Shell Sport

B-Tech Computer Science & Engineering
Syllabus for Semester-V

Course Code	Course Title	Hours Per Week		Credit
		L	P	
CSE301C	Design and Analysis of Algorithms	3	0	3
CSE302C	Operating System	3	0	3
CSE303C	Database Management System	3	0	3
CSE304C	Computer Networks	3	0	3
CSE305C	Numerical Methods	3	0	3
CSE306C	Simulation Lab	1	4	3
CSE307C	Design and Analysis of Algorithms Lab	0	2	1
CSE308C	Database Management System Lab	0	2	1
CSE309C	Computer Networks Lab	0	2	1
Total Credits		**16**	**10**	**21**

Course Code: **CSE301C**	Course Title: **Design and Analysis of Algorithms**	Credits: 03 **L - 3 P - 0**

Course Outcomes:

- *Understand and use asymptotic notations to analyze the performance of algorithms*
- *Understand and analyze the design of algorithms using Brute force, Divide & Conquer, Dynamic Programming, Greedy technique, Backtracking, Branch & Bound techniques.*
- *Compare and contrast various search and sorting techniques.*
- *Apply the various algorithms to solve problems and analyze their efficiency.*

Unit - I

Introduction: Algorithm Design paradigms- motivation, Concept of algorithmic efficiency, Run time analysis of algorithms, Asymptomatic Notations.

Divide & Conquer: Structure of divide and conquer algorithms: examples, Binary search, Merge Sort, Quick sort, Analysis of divide and conquer run time. Recurrence Relations, Master Theorem for solving Recurrence Relations.

Unit - II

Greedy method: Overview of the greedy paradigm, examples of exact optimization solution (minimum cost spanning tree), approximate solution (Knapsack problem), Huffman coding, Single source shortest path.

Unit - III

Dynamic Programming: Overview, difference between dynamic programming and divide and conquer, applications: Shortest Path in Multistage Graph, Non-fractional (0/1) Knapsack problem, Matrix Chain Multiplication, Travelling salesman problem, Longest common sequence.

Unit - IV

Graph searching and traversal: Overview, traversal methods, depth first and breadth first search. Dijkstra's and Bellman Fort Algorithm for finding Single source shortest paths. All pair shortest paths and matrix multiplication, Floyd – Warshall algorithm for all pair shortest paths.

Unit - V

Back Tracking: Overview, 8-queen problem and Knapsack problem.

Branch & Bound: LC searching, bounding, FIFO branch and bound, Applications: 0/1 Knapsack problem, Travelling salesman problem.

Computational complexity: Complexity measures, Polynomial vs non-polynomial time complexity; NP hard and NP complete classes, examples

Books Recommended:

1. T. H. Cormen, C. E. Leiserson, R. L. Rivest, Clifford Stein, "*Introduction to Algorithms*", 2nd Ed., PHI, 2004.

References Books:

- Ellis Horowitz and Sartaz Sahani, "*Computer Algorithms*", Galgotia Publications, 1999.
- V. Aho, J. E. Hopcroft, J. D. Ullman, "*The Design and Analysis of Computer* Algorithms", Addition Wesley, 1998.
- D. E. Knuth, "*The Art of Computer Programming*", 2nd Ed., Addison Wesley, 1998.

Course Code: **CSE302C**	Course Title: **Operating System**	Credits: 03 **L - 3 P - 0**

Course Outcomes:

- *Describe and explain the fundamental components of a computer operating system.*
- *Describe the various types and models of operating systems.*
- *Define, restate, discuss, and explain the policies for scheduling, deadlocks, memory management, synchronization, system calls, and file systems.*
- *Describe and extrapolate the interactions among the various components of computing systems.*
- *Describe and simulate the following OS components: System calls, Schedulers, Memory management systems, Virtual Memory and Paging systems.*

UNIT - I

Operating system overview – Objectives, structure and functions, Evolution of Operating System, Memory Hierarchy, I/O techniques, Multiprocessor and Multicore Organization.

UNIT - II

Process Management: Process concept, Process States, Concurrent processes, process control block, Inter-process Communication.
Threads: Processes and Threads, Types of Threads, Multithreading models.
Process Scheduling: Schedulers, Scheduling criteria, Scheduling algorithms.

UNIT - III

Process Synchronization: Critical Section Problem, Peterson's solution, Mutual Exclusion, Semaphores, Bounded Buffer Problem, Readers/Writers problem, Monitors.
Deadlocks: characterization, prevention, avoidance, detection, Recovery.

UNIT - IV

Memory Management: Overview, Swapping, Contiguous Memory Allocation, Paging and Segmentation.
Virtual Memory Management: Overview, Demand Paging, Page Replacement, Thrashing.

UNIT - V

I/O Management & Disk Scheduling: I/O devices and organization of I/O function, I/O Buffering, DISK I/O, Operating System Design Issues.
File System: File Concept, File Organization and Access Mechanism, File Directories, File Sharing, Implementation Issues.

Text Books:

1. Silberschatz, Peter Galvin, Greg Gagne "*Operating System Principles*", 9th edition.

Reference Books:

1. Andrew S. Tannenbaum & Albert S. Woodhull, "*Operating System Design and Implementation*", Prentice Hall.
2. William Stallings, "*Operating Systems – internals and design principles*", Prentice Hall.
3. Andrew S. Tannenbaum, "Modern Operating Systems", Prentice Hall.
4. Gary J.Nutt, "*Operating Systems*", Pearson/Addison Wesley.
5. Pramod Chandra P.Bhatt, "*An Introduction to Operating Systems Concepts and Practice*".

Course Code: **CSE303C**	Course Title: **Database Management System**	Credits: 03 **L - 3 P - 0**

Course Outcomes:

- *Describe relational schema design, and it covers the normalization and functional dependency algorithm.*
- *Understands the relational algebra concepts, selection, projection, relational calculus which helps in understanding queries.*
- *Understand various transaction processing, concurrency control mechanisms and database protection mechanisms.*

Unit 1: Introduction

Basic Concepts and Conceptual Database Design, Characteristics of the Database, Data Models, DBMS Architecture, Data Modelling Using the Entity-Relationship Model, Translating ER Model into Relational Model, Extended Entity-Relationship Model.

Unit 2: Relational Database Design

Functional Dependencies, Normalization – 1NF, 2NF, 3NF, BCNF, higher Normal forms, Lossless Join & Dependency Preserving Decomposition, De-normalization.

Unit 3: Relational Data Model

Relational Algebra Basic Operators, Composition of Operators, Additional Operators, Extended Relational Algebra, Tuple calculus, SQL-Introduction and Data Definition, Basic Queries, Advanced Queries, Updates, Joins, Views and Triggers, Brief introduction of PL-SQL.

Unit 4: Transaction and Concurrency Control

Transaction- Concept and purpose, ACID properties and their necessity, Problems with full isolation, Concurrency Control- lock-based protocols, 2-phase locking, Timestamp-based Protocols, Validation–based Protocols, Multiple Granularity, deadlock handling.

Unit 5: Recovery:

Failures and their classification, recovery and atomicity, recovery algorithms, Undo-Redo with write ahead logging, Buffer management, File organization, indexing (e.g., B and B+ trees), Hashing and Query Processing

Books Recommended:

1. Elmsari and Navathe, "*Fundamentals of Database Systmes*", A. Wesley
2. Korth, Silberschatz, "*Database System Concepts*", TMH
3. Steve Bobrowski, "*Oracle 8 Architecture*", TMH
4. Date C. J., "*An Introduction to Database Systems*", Narosa Publishing
5. Ullman J. D., "*Principles of Database Systems*", Galgotia Publications
6. William Page, "*Using Oracle 8i – Special Edition*", Que/PHI
7. Ivan Bayross, "*SQL & PL/SQL Using Oracle 8i & 9i with SQLJ*", BPB

Course Code: **CSE304C**	Course Title: **Computer Network**	Credits: 03 **L - 3 P - 0**

- *Understand the networking problem and its solution and understand the essence of layering mechanism of OSI and TCP/IP network models.*
- *Explore the data link layer and understand the need of various protocols for efficient communication.*
- *Understand the addressing mechanism of internet and protocols which help smooth functioning of the today's congested internetworked world.*
- *Understand the security implications in the modern internet and how they are resolved.*

UNIT – I

Introduction: Uses of Computer Networks, Network and Protocol Architecture, Reference Model (ISO-OSI, TCP/IP-Overview), Types of networks (LAN, MAN and WAN), Different network topologies like star, ring, hybrid, tree etc. IEEE standards.

UNIT -II

Data Link layer – Design Issues, Error detection and Correction techniques, Flow control algorithms, Framing techniques, congestion control.

UNIT - III

Network Layer: IP Addressing, IPv.4, IP protocols, sub-netting, Different Routing algorithm, Congestion control Algorithms General principles of congestion prevention policies.

UNIT - IV

Internetworking: Internet control protocols: ICMP, ARP, RARP, BOOTP, DHCP, Gateway routing protocols: OSPF, BGP.

Transport Layer: Transport services, Elements of Transport protocols, Internet Transport Protocols. TCP, UDP, TCP sockets

UNIT – V

Application Layer-Network Security: Domain Name system (DNS): Electronic Mail; The world Wide Web. Multimedia.

TEXT BOOKS:

1. W. Stallings, "*Computer Communication Networks*", PHI, 1999.
2. Larry L.Peterson, Peter S. Davie, "*Computer Networks*", Elsevier, Fifth Edition, 2012.
3. *Computer Networks* – Third Edition – Andrew S. Tanenbaum, Prentice Hall of India.
4. *Data Communications and Networking* – Behrouz A. Forouzan, Third Edition TMH.

References:

1. U. Black, "*Computer Networks-Protocols, Standards and Interfaces*", PHI, 1996.
2. Laura Chappell, "*Introduction to Cisco Router Configuration*", Techmedia, 1999.
3. Michael A. Miller, "*Data & Network Communications*", Vikas Publication, 1998.
4. William A. Shay, "*Understanding Data Communications & Networks*", Vikas Publication, 1999.

Course Code: **MTH-331C**	Course Title: **Numerical Methods**	Credits: 03 **L - 3 P - 0**

Course Outcomes:

- *Understand the networking problem and its solution and understand the essence of layering mechanism of OSI and TCP/IP network models.*
- *Explore the data link layer and understand the need of various protocols for efficient communication.*
- *Understand the addressing mechanism of internet and protocols which help smooth functioning of the today's congested internetworked world.*
- *Understand the security implications in the modern internet and how they are resolved.*

Unit - I:

Finite Difference: Difference Table and its usage. The difference operators Δ, ∇ and the operator E.

Interpolation: Interpolation with equal intervals, Newton's advancing difference formula. Newtons's backward difference formula. Interpolation with unequal intervals. Newton's divided difference formula. Lagrange's interpolation formula.

Central Differences: The central difference operator δ and the overraging operator μ. Relations between the operators. Gauss forward and backward interpolation formula, Sterlings, Bessel's, Laplace and Everetts formulae.

Unit - II

Inverse interpolation: Inverse interpolation by (i) Langrange's (ii) Methods of successive e approximation & (iii) Methods of elimination of third differences

Numerical solution of algebraic and Transcendental Equations: Graphic Method, Regula-Fast method, Balzano's Process of bisection of intervals, Newton-Raphson Method and its geometrical significance.

Unit - III:

Numerical Integration: Numerical Integration, General Quadrature Formula, Simpson's one-third and three-eight rules, Weddles' rule, Hardy's rule, Trapezoidal rule.

Numerical differentiation: Numerical differentiation of a friction. Differential coefficient of a friction in terms of its differences. Applications

Unit - IV

Difference Equations: Linear-homogeneous and non-homogeneous difference equations of order n with constant coefficient, and their solution, methods of undetermined coefficient.

Numerical Solution of ordinary differential equations: Numerical solution of ordinary differential equations, Picard's method. Taylors series method, Euler's method, Runge-Kutta Method.

Books Recommended:

1. *Numerical Methods for Scientists and Engineering*, by M.K.Jain, S.R.Iyengar & R.K. Jain, Wiley Eastern Ltd.
2. *Mathematical Numerical Analysis*, by S.C. Scarborough, Oxford and IBH.
3. *Introductory methods in Numerical Analysis*, by S.S.Sastry, Prentice Hall of India**.**
4. *Numerical Solution of Differential equations*, by M.K.Jain.
5. *Numerical Methods for Science & Engineering*, by R.G.Stanton.

Course Code: **CSE306C**	Course Title: **Simulation Lab**	Credits: 03 **L - 1 P - 4**

Course Outcomes:

- *Understand the concept of simulating the real-world problems through software tools.*
- *Understand the basics of MATLAB and apply the methods of MATLAB for problem solving.*
- *Understand the basics of SCILAB and apply them for problem solving.*
- *Design and simulate various network topologies used in computer networks.*

UNIT – I

Matlab: Introduction to MATLAB, Identify and use various windows, Data types, Rules about variable names, Express numbers in either floating-point or scientific notation, save a series of commands. Built in functions, elementary math functions (common math functions, rounding functions, discrete mathematics functions, trigonometric functions), data analysis functions (maximum and minimum, mean and median, sums and products), sorting functions, random numbers. Relational and Logical Operators, If-else statements, Switch-case statements, For loop, While loop, Special commands (Break and continue), Script file. Defining matrices, accessing matrix elements, Matrix Operations and Functions. Two-Dimensional Plots, Three Dimensional Plots, Saving Your Plots.

UNIT – II

Scilab: Introduction to Scilab, Identify and use various windows, Data types, Rules about variable names, Express numbers in either floating-point or scientific notation, save a series of commands. Built in functions, elementary math functions (common math functions, rounding functions, discrete mathematics functions, trigonometric functions), data analysis functions (maximum and minimum, mean and median, sums and products), sorting functions, random numbers. Relational and Logical Operators, If-else statements, Switch-case statements, for loop, While loop, Special commands (Break and continue), Script file. Defining matrices, accessing matrix elements, Matrix Operations and Functions, Two-Dimensional Plots, Three Dimensional Plots, Saving Your Plots.

UNIT – III

Cooja: Introduction to cooja, Running Cooja Simulator, creating a new simulation, creating a new mote type, adding motes and running the simulation, saving simulation file, Debugging with Cooja Simulator.

UNIT – IV

Ns2: Introduction to Open Source Software, Introduction to Network Simulator – 2 (NS2), TCL Scripting, TCL Script Components, Introduction to NAM, TRACE, XGRAPH etc., Wired Scenarios - Writing your first TCL Script, Network Dynamics etc., Simulation of Different Network Topologies – BUS topology, RING Topology, STAR topology etc. Creating Wireless Scenarios, Wired-cum-Wireless Scenarios etc.

Text Book

1. Holly Moore, "MATLAB for Engineers", Pearson
2. Bansal R.K, Goel A.K., Sharma M.K., "*MATLAB and its Applications in Engineering*", Pearson Education, 2012.
3. Baudin, Introduction to Scilab, Scilab Consortium

References

2. Stephen.J.Chapman, *"Programming in MATLAB for Engineers*", Cengage Learning, 2011.
3. Gomez, Engineering and scientific computation with Scilab, Springer

Course Code: **CSE307C**	Course Title: **Design and Analysis of Algorithms Lab**	Credits: 01 **L - 0 P - 2**

Course Outcomes:

- *Understand and use asymptotic notations to analyze the performance of algorithms*
- *Understand and analyze the design of algorithms using Brute force, Divide & Conquer, Dynamic Programming, Greedy technique, Backtracking, Branch & Bound techniques.*
- *Compare and contrast various search and sorting techniques.*
- *Apply the various algorithms to solve problems and analyze their efficiency*

Lab Details:

The Following Problems Are to Be Solved In C

- Simple Experiments on time and space complexity of a program
- **Divide and Conquer Approach:** Binary Search, Merge Sort, Quick sort, Medians and Order statistics,
- **Greedy Algorithms:** Knapsack Problem, Huffman Codes.
- **Dynamic Programming:** Matrix Chain Multiplication, Traveling Salesman Problem, Longest common subsequence, 0/1 Knapsack
- **Graph Algorithms:** Representation of Graphs, Breadth First Search, Depth First Search, Topological Sort, Strongly Connected Components, Algorithm for Kruskal's and Prim's for finding Minimum cost Spanning Trees, Dijkstra's and Bellman Fort Algorithm for finding Single source shortest paths. All pair shortest paths and matrix multiplication, Floyd – Warshall algorithm for all pair shortest paths.
- **Back Tracking:** 8-queen problem and Knapsack problem.
- **Branch & Bound:** LC searching, bounding, FIFO branch and bound

Course Code: **CSE308C**	Course Title: **Database Management System Lab**	Credits: 01 **L - 0 P - 2**

Course Outcomes

- *Design and Implement a database schema.*
- *Devise queries using DDL, DML, DCL and TCL commands.*
- *Develop application programs using PL/SQL.*
- *Design and implement a project using embedded SQL and GUI.*
- *Designing Forms and Reports.*

Lab Details:

1. SQL Data Definition and Data Types.
2. Specifying Basic Constraints in SQL.
3. Schema change statements in SQL.
4. Insert, Delete and Update Statements in SQL.
5. Basic Queries in SQL.
6. Joining Multiple Tables: Inner Joins, Outer Joins, Cross Joins.
7. Correlated and Nested Queries in SQL.
8. Implement all the operation like Union, Intersect, minus, in, exist, aggregate and Statistical
9. Functions, Group By And Having clause.
10. Creating Views, Writing Assertions.

PL/SQL

1. PL/SQL data Types, Control Structures, Logical Operators (And, Not, Or), Relational Operators (IS NULL, Like, Between, In etc).
2. Built in Functions.
3. Functions and procedures.
4. Cursors, Collections and records.
5. Packages and Triggers.
6. Error handling.
7. Oracle administration
8. Creating FORMS
9. Generating REPORTS.

Course Code: **CSE309C**	Course Title: **Computer Networks Lab**	Credits: 01 **L - 0 P - 2**

Course Outcomes:

- *Students should be able to understand and identify the cabling technologies.*
- *Students should be able to connect two computers peer to peer via switch and router*
- *Students should be manually able to provide IP addresses to machines without DHCP.*
- *Should be able to share resources via network and configure windows server and DHCP.*
- *Should be able to use network simulators like NS2, OPNET and GLOMOSIM and should be able to run wired and wireless topologies in NS2.*

Lab Details:

1. Identification of guided media (UTP, Fiber) /Color coding.
2. To Connect two pc using peer to peer communication/via switch/via router.
3. IP addressing (static and dynamic).
4. Sharing the resources in wired network (software and hardware).
5. Configuring the Windows server (Active directory) and DHCP server.
6. Study of NS2/ GLOMOSIM / OPNET / QUALNET.
7. To implement wired network topology and wireless network topology in NS2 or any other simulator.

B-Tech Computer Science & Engineering
Syllabus for Semester-VI

Course Code	Course Title	Hours Per Week		Credit
		L	P	
CSE350C	Microprocessors, Peripherals and Interfacing	3	0	3
CSE351C	Formal Language and Automata Theory	3	0	3
CSE352C	Artificial Intelligence	3	0	3
CSE353C	Java Programming	3	0	3
	Elective- I	3	0	3
CSE354C	Microprocessors, Peripherals and Interfacing Lab	0	2	1
CSE355C	Java Programming Lab	0	2	1
CSE356C	Project I	0	2	1
CSE357C	Industrial Training	0	0	1
Total Credits		**15**	**6**	**19**

Course Code: **CSE350C**	Course Title: **Microprocessors, Peripherals and Interfacing**	Credits: 03 **L - 3 P - 0**

Course Outcomes:

- *To have a complete knowledge of architecture of micro-processor 8085, its pin layout and description of all signals.*
- *To understand the instruction cycle, timing diagrams, registers, flags etc. of 8085 micro-processor in detail.*
- *To have a knowledge of instructions, addressing modes, interrupts, subroutines, conditional call instructions of 8085 and be able to perform assembly language programming for numerous operations in 8085.*

UNIT - I

Introduction To 8-bit Microprocessor: history of microprocessor, microprocessor instruction set basics and memory concept. 8085 programming model, registers, flags, buses.
Architecture of 8-bit Microprocessor: Intel 8085A Microprocessor, Pin Description and Internal Architecture of 8085. Instruction Classification: 1-byte, 2-byte and 3-byte instructions

UNIT - II

Intel 8085 Microprocessor Instruction Set and Programming: Addressing modes of 8085, Instruction Classification: data transfer, arithmetic, logical, rotate, branch and machine control instructions. Development of 8085 assembly language programs using these instructions. Operation and Control of Microprocessor: timing and control unit, concepts of instruction cycle, machine cycle and t-states. op-code fetch machine cycle, memory read/write machine cycles, i/o read/write machine cycles, time delays.

UNIT - III

Counters, time delays and related examples. Stacks, subroutines and related instructions for their implementation. Interrupts: Interrupt structure of 8085A microprocessor, processing of vectored and non-vectored interrupts, instructions related to interrupts

UNIT - IV

Basic interfacing concepts. Memory interfacing, address allocation technique and decoding. Interfacing output displays and input devices, memory mapped I/O. 8255A Programmable Peripheral Interface. The 8254 Programmable Interval Timer. Brief description of 8259A Programmable Interrupt Controller. Direct Memory Access and 8237 DMA Controller

UNIT - V

Introduction to Advanced Microprocessors: Intel 8086 Microprocessor architecture, addressing modes, pin configuration & function of each pin. Introduction and advance features of 8088, 80186, 80286, 80386 and 80486 microprocessors. Latest developments in microprocessor technology.

RECOMMENDED BOOKS:

1. *Microprocessor Architecture, Programming, and Applications with the 8085* – Ramesh S. Gaonkar, Pub: Penram International.
2. Hall D.V.,"*Microprocessor and Interfacing-Programming and Hardware*", 2nd Ed., Tata McGraw-Hill Publishing Company Limited, 2008.
3. *8085 Microprocessor And its Applications*, by A. Nagoor Kani, Third Edition, TMH Education Pvt. Ltd

Course Code: **CSE351C**	Course Title: **Formal Languages and Automata Theory**	Credits: 03 **L - 3 P - 0**

Course Outcomes

- *Explain and manipulate the different concepts in automata theory and formal languages.*
- *Explain the power and the limitations of regular languages and context-free languages.*
- *Discussing applications of automata theory in a variety of contexts.*
- *Develop an understanding of computation through Turing Machines.*
- *Establish foundations for exploring more advanced topics like compiler design*

Unit – I

Introduction to finite Automata: Basic Terminology: Alphabet**,** Introduction to finite automata, Deterministic finite automata, Nondeterministic finite Automata, Equivalence of DFA and NDFA, Non-Deterministic Finite automata with Λ-Transitions. application of finite Automata

Unit - II

Regular Expressions: Regular Languages and Regular Expressions, Arden's Method.
Properties of Regular sets: The Pumping Lemma for Regular sets, Application of Pumping Lemma, Closure Properties of Regular Sets, Minimization of Finite Automata, Minimization Algorithm.

Unit - III

Context Free Grammars and Languages: Context free grammars, Parse trees, Ambiguity in Grammars, Left Recursion, Eliminating Epsilon Productions, Eliminating Unit productions, Chomsky Normal Form, Grieback Normal Form

Unit - IV

Pushdown Automata: Definition of the pushdown automata, the languages of a PDA, Equivalence of PDA's and CFG's, Deterministic Pushdown Automata

Unit - V

Introduction to Turing Machine: The Turing Machine, Programming techniques for Turing machines. **More General Languages and Grammars:** Recursively Enumerable and Recursive Languages, Unrestricted Grammars, Context Sensitive Language and Grammar. Relation between Languages of classes, Chomsky hierarchies of grammars.

Text Books:

1. Jhon E. Hopcroft, Rajeev Motwani, Jeffery Ullman: *Introduction to Automata Theory, Languages and Computation*, 3rd edition, Pearson education, 2007
2. Martin J.C., "*Introduction to Languages and Theory of Computation*", 3rd Edition, Tata McGraw-Hill Publishing Company Limited.

Reference Books:

1. K.L.P. Mishra: *Theory of Computer Science, Automata, Languages and Computation*, 3rd edition, PHI, 2007
2. Jhon C martin: *Introduction to languages and Automata Theory*, 3rd edition, Tata McGraw-hill,2007
3. A.M. Padma Reddy: *Formal Languages and Automata theory*, 3rd edition, United Publishers,2009
4. Shirish S. Sane: *Theory of computer science*, 2nd edition, Technical publications Pune

Course Code: **CSE-711T**	Course Title: **Artificial Intelligence**	Credits: 03 **L - 3 P - 0**

Course Outcomes

- *Identify and formulate a problem as an AI problem*
- *Learn the various searching techniques and apply them to solve problems.*
- *Understand knowledge-based systems and ability to design them.*
- *Ability to design an expert system.*

UNIT - I: INTRODUCTION

Introduction to AI and intelligent agents, Characteristics of AI, Applications of AI, Turing Test, Solving Problems by Searching, Uninformed Search, Depth first Search, Breadth first Search.

UNIT - II: SEARCH TECHNIQUES

Heuristic search techniques, Hill climbing, Branch and Bound technique, Best first search, A* algorithm, AND/OR Graphs, Problem reduction and AO* algorithm, Game Playing Min Max Search procedure, Alpha-Beta cut-off.

UNIT - III: KNOWLEDGE AND REASONING

Building a Knowledge Base- Propositional logic, First Order Predicate Calculus, Skolemnisation, Resolution Principle, Inference Mechanism, Semantic Networks, Frame System.

UNIT - IV: EXPERT SYSTEMS

Introduction to Expert Systems, Architecture of Expert Systems, fuzzy logic and its applications, Basic Probability Notation, Baye's Rule and Its Use.

UNIT - V: PLANNING

Forward/Backward Search, Goal Stack Planning, Sussman's Anomaly.

Text Books

1. Elaine Rich and Kevin Knight: Artificial Intelligence – Tata McGraw Hill.

Reference Books

1. Char nick "*Introduction to Artificial Intelligence*", Addision Wesley.
2. Dan W.Patterson, *Introduction to Artificial Intelligence and Expert Systems* – Prentice Hal of India.
3. Winston, "*LISP*", Addison Wesley.
4. *Artificial Intelligence: A Modern Approach*, Stuart Rusell, Peter Norving, Pearson Education 2nd Edition.

Course Code: **CSE353C**	Course Title: **Java Programming**	Credits: 03 **L - 3 P - 0**

Course Outcomes

- *Able to understand power of Object-Oriented Programming knowledge and gain insight in the structure and model of the Java programming language.*
- *Demonstrate basic problem-solving skills: analyzing problems, modeling a problem as a system of objects and implementing models in Java*
- *Evaluate user requirements and decide whether the Java programming language can be used/prove as an efficient tool in meeting user requirements*
- *Develop software in the Java programming language (application)*

Unit – I: Introduction to Java

Importance of java to internet, Java Virtual Machine Architecture, Class loading process by Class loaders, Role of Just in Time Compiler (JIT) , Data Types, Variables ad Arrays ,precedence and associability rules, Operators, Control statements, Object oriented Paradigms Class Fundamentals, Command Line Arguments, static initializer, Inheritance, abstract classes, Variable Hiding, Overloading and Overriding of Methods , Access Controls modifiers, Nested and Inner Classes, Dynamic method dispatch introduction to Packages, Understanding CLASSPATH, Access Protection, Importing Packages, Distributing packages as ZIP and Jar files, Defining and implementing an Interface, Abstract classes Vs Interfaces.

Unit - II: Fundamentals of Exception handling

Types of exceptions, exception handlers, Try and catch, Multiple catch clauses, Nested try statements, throw, throws and finally, Creating custom exceptions String class, String buffer class, String builder class, Wrapper class. File Handling, Binary Streams, Character Streams, Serialization. Lambda Expressions: Introduction to Lambda Expressions, Using Lambda Expression, Method References

Unit - III: Windows Programming

Introduction to AWT, containers and components, AWT classes, AWT controls, Layout managers and Menus Event Delegation Model: Event Classes Event Listeners Swing: Introduction to JFC Controls. Applet: Applet Basics, Architecture and Skelton, Simple Applet Display Methods.

Unit – IV: The Java Thread Model

Creating a Thread: Extending Thread, Implementing Runnable, Creating Multiple Threads and Context Switching, Synchronization, Inter-thread Communication, Java collection framework, maps and Generics, Introduction to Stream API, Steam API Examples

Unit - V RMI: Distributed Applications:

RMI Architecture, Implementation of RMI Server and Client, Call-back Mechanism JDBC: JDBC Drivers, JDBC API, Executing statements, prepared statements and callable statements, Metadata, Scrollable & Updatable Result Set, Data Source & Connection Pooling, Batch Updates, Row sets

Text Books/References

1. *"The Java Handbook",* by Patrick Naughton, Michael Morrison Osborne/McGraw-Hill publication
2. *"Java 2 The Complete Reference",* by Herbert Schidlt Tata McGraw-Hill publication.
3. *"Professional Java Server Programming J2EE edition Volume I, II",* Wrox publications
4. *"SCJP Sun certified programmer for java 6 exam study guide",* by Kathy Sierra and Bert Bates Dreamtech press
5. *"Head First Java",* by Kathy Sierra and Bert Bates oreilly publications
6. "Horstmann", "*Computing Concepts with Java 2 Essentials",* John Wiley
7. "*Programming with Java*", by E Balaguruswamy.
8. Decker & Hirshfield, "*Programming Java*", Vikas Publication.
9. *"Java: An Introduction to Computer Science and Programming*", by W.Savitch (Prentice-Hall).
10. *"Advanced Java: Internet Applications",* by A.Gittleman (Scott Jones).

Course Code: **CSE354C**	Course Title: **Microprocessors, Peripherals and Interfacing Lab**	Credits: 01 **L - 0 P - 2**

Course Outcomes

- *To get the thorough knowledge of micro-processor trainer kit, how to use it, specifications etc.*
- *To be able to write the assembly language programs in micro-processor.*
- *To get familiarized with the instruction set and addressing modes of 8085.*
- *To be able to interface A/D, D/A convertors, stepper motors etc. to 8085.*

Lab Details:

1. Write a program to add two 16-byte numbers.
2. Write an assembly language program to swap two 8-bit numbers stored in 8085 microprocessors.
3. Write a program for adding first N natural numbers and store the results in memory location X.
4. Write a program to find maximum/minimum of two 8-bit numbers.
5. Write an assembly language program to find maximum and minimum of 10 numbers.
6. Write a program to count odd/even numbers in series of 10 numbers.
7. Write a program to find the sum of series of even numbers.
8. Write a program to count number of l's in the contents of D register and store the count in the B register.
9. Write a program to sort given 10 numbers from memory location 2200H in the ascending order.
10. Write a program to multiply two single byte unsigned numbers.
11. Write a program to multiply two 8-bit numbers using logical instructions.
12. Write a program to multiply two positive floating-point numbers.
13. Write a program to divide a 4-byte number by another 4 byte number.
14. Write a program to introduce a time delay of 100 ms. Using this program as a subroutine, display numbers from 01H to 09H with the above calculated time delay between every two numbers.
15. Write a 20 ms time delay subroutine using register pair BC. Clear the Z flag without affecting any other flags in the flag register and return to the main program
16. Interface a display circuit with the microprocessor either directly with the bus or by using I/O ports. Write a program by which the data stored in a RAM table is displayed.
17. To design and interface a circuit to read data from an A/D converter, using 8255A in the memory mapped I/O.
18. To design and interface a circuit to convert digital data into analog signal using 8255 A in the memory mapped I/O.
19. To interface a keyboard with the microprocessor using 8279 chip and transfer the output to the printer.
20. To design a circuit to interface a memory chip with microprocessor with given memory map.

Course Code: **CSE356C**	Course Title: **Project I**	Credits: 01 **L - 0 P - 2**

Description:

During sixth semester, students have to take Project – I (Mini – Project) of one credit. In Project – I, students need to identify area of their interest in which they would opt (not necessarily) their Project – II and Project – III later in 7^{th} and 8^{th} semesters respectively. In Project – I, students have to do extensive literature survey of their field of interest and also learn certain tools / software required for Project – II and Project – III accomplishment.

Course Code: **CSE357C**	Course Title: **Industrial Training**	Credits: 01 **L - 0 P - 0**

Description

Students are required to undergo Industrial Training for at least for 4 – weeks duration. It is a one credit course and should be completed by or before 5th semester. During this training students join some reputed institutes / industries with prior permission from the department. The purpose of Industrial Training is to make students aware about cutting-edge technologies and industry readiness for their future endeavors.

B. Tech Computer Science and Engineering
Syllabus of Discipline Centric Electives for
Semester Sixth

Course Code	Course Title	Hours Per Week		Credits	Pre-requisite	Semester
		L	P			
CSE350E	Computer Graphics	3	0	03	Computer Programming	6th
CSE351E	Introduction to Core Java Script	2	2	03	Internet and Web Designing	6th
CSE352E	Software Testing and Quality Assurance	3	0	03	Software Engineering	6th
CSE353E	C# and .Net Programming	2	2	03	Computer Programming	6th
CSE354E	Distributed Computing	3	0	03	OS	6th
CSE355E	Digital Image Processing	3	0	03	Computer Programming	6th
CSE356E	Programming in in R	2	2	03	Nill	6th
CSE357E	Soft Computing	3	0	03	Computer Programming	6th

Course Code: **CSE350E**	Course Title: **Computer Graphics**	Credits: 03 **L - 3 P - 0**

Course Outcomes

- *Know and be able to discuss hardware system architecture for computer graphics.*
- *Know and be able to use a current 3D graphics API (e.g., OpenGL or DirectX) graphics pipeline, frame buffers, and graphic accelerators/co-processors.*
- *Students will write program functions to implement graphics primitives.*
- *Students will write programs that demonstrate geometrical transformations.*

UNIT - I

Introduction to Graphics, Manual drafting vs. Computer Graphics, Advantages of Computer Graphics, Characteristics & Applications of Computer Graphics, Graphic Display Devices (CRT, Random Scan Display monitors, Raster Scan Display monitors). Graphic Input Devices (Keyboard, Mouse, Trackballs & Space balls, Joy-sticks, Touch-screens).

UNIT - II

Points & Lines, Line Drawing Algorithms (DDA, Bresenham's Line Algorithm). Circle – Generating Algorithms (Basic concepts & properties of circle drawing, Mid-pt. circle algorithm). Ellipse Generating Algorithms (Basic concepts & properties of ellipse drawing, Mid-pt. Ellipse-algorithm). Boundary Filling Algorithm, Flood Filling Algorithm, Aliasing / Anti-aliasing.

UNIT - III

Introduction to 2D- transformation: Basic transformation (Translation, Rotation & Scaling). Composite Transformations (Translation, Rotation & Scaling). Other Transformations (Reflection & Shearing).
2-D Viewing: The viewing pipeline, Clipping operations, Point Clipping, Line clipping (Cohen –Sutherland Line Clipping). Polygon clipping (Sutherland-Hodgeman Polygon Clipping), Text clipping.

UNIT - IV

Projections (Parallel Projections & Perspective Projections), Visible-Surface Detection Methods (Classification of Visible –Surface Detection Algorithms Z-buffer algorithm, Scan line algorithm, Area subdivision algorithm, Ray tracing algorithm, Painter's Algorithm).

UNIT - V

Object Rendering, Introduction Object-Rendering, Light Modelling Techniques, illumination Model, Shading, Flat Shading, Polygon Mesh Shading, Gouraud Shading Model, Phong Shading, Interactive Picture –Construction Techniques (Basic Positioning Methods, Constraints, Grids, Gravity field, Rubber Band Techniques, Dragging, Painting & Drawing, Inking).

Text Books:

1. *"Computer Graphics*", by Donald Hearn & M. Pauline Baker.

Reference Books:

- "*Principles of Interactive Computer Grap*hics", by William. M. Newman & Robert. F. Sproull.
- Steven Harrington, "*Computer Graphics A Programming Approach*", McGraw Hill.
- James.D. Foley, VanDam, "*Fundamentals of interactive Computer Graphics*".
- David F. Frogers & J Alan Adams, "*Procedure and elements of Computer graphics*".

Course Code: **CSE351E**	Course Title: **Introduction to Core JavaScript**	Credits: 03 **L - 2 P - 2**

Course Outcomes

- *demonstrate and describe the function of Java Script in Programming.*
- *Apply the best practices of Java Script such as DOM/Interactivity with elements.*
- *examine frameworks such as JQuery and learn the basics of PHP programming.*
- *experiment with database and create basic action script coding.*

UNIT - I

Fundamentals: Variables and Flow Control Variables and constants, Rules for naming variables, Data types, Numbers, Arithmetic Operations, Strings, Booleans and Comparison Operators, Conditional Statements (if, else if, else), Logical AND and Logical OR Operators, Variable Scopes, Hoisting, Strict Mode.

UNIT - II

JavaScript Functions, Arrays, Loops. Introduction to Functions, Undefined and Null Data Types, Arguments, Multiple Arguments and Arguments Defaults, Function Scopes. Project: Build a Grade Calculator (total marks entered generates a message). Array basics (creating array and reading from array), Manipulating Arrays with Methods, Looping over Arrays, Searching Arrays, Filtering Arrays, Sorting Arrays. Project: Build a Todo List using Arrays.

UNIT - III

Objects and DOM Manipulation Object Basics, Object Properties, Creating Objects, Read and Modify Objects, Object Methods. DOM Manipulation, targeting page elements using JS, adding new elements via DOM [Dynamic Rendering of Pages], Handling User Interactions (event listeners), Text Inputs and Data Filtering, Rendering Filtered Data to page.
Project: GUI based Graphical To-do List.

UNIT - IV

Expanded JavaScript, Data Storage and ES6 Arrow Functions, Type Coercion, Catching and Throwing Errors. Introduction to Regular Expressions. Saving Data into Local Storage using JS, reading saved Data from Local Storage, Modifying LocalStorage Values. Difference between var and let, Array.find(), Array.findIndex( ) .

UNIT - V

JavaScript Object Notation JSON JSON Syntax, JSON Data Types, JSON Parse, JSON Stringify, JSON Objects, JSON Arrays. Introduction to JSONP.
PROJECT: Build a Web Application that makes use of JavaScript covered in the course. This Application should be user interactive and not static.

Books

1. *JavaScript and JQuery: Interactive Front–End Web Development* – Jon Duckett
2. *JavaScript & jQuery: The Missing Manual, 3rd Edition* - David McFarland

References : https://www.w3schools.com/js/default.asp (w3 Schools) Mozilla Developer Network (MDN)

Lectures : https://wsww.edx.org/course/javascript-introduction (JavaScript Introduction, edX)

Course Code: **CSE352E**	Course Title: **Software Testing and Quality Assurance**	Credits: 03 **L - 3 P - 0**

Course Outcomes

- *Design and implement various testing strategies to improve software quality.*
- *An overview of more advanced testing methods (such as model-based testing, mutation testing and search-based testing), and in the state-of-the-art in software testing research.*
- *Have a working knowledge and experience in static and dynamic code analysis.*

UNIT - I

Software Testing: Testing as an Engineering Activity, Role of Process in Software Quality, Testing as a Process, Software Testing Principles, Tester Role in Software Development, Artifacts of testing (Faults, Errors, and Failures), Limitations of Testing, Challenges in Software Testing, Testing and debugging, Verification, Validation, Test levels.

Software Quality: Software Quality, Software Control, Quality Assurance, Quality Assurance Analyst, Quality Factor, Quality Management, Methods of Quality Management, Core components of Quality, Cost Aspect of Quality.

UNIT - II

Different Testing Techniques, Differences between testing techniques

Black Box Testing: Requirements based testing techniques, Boundary value analysis, Equivalence partitioning, Decision table, State/Graph based testing

White Box Testing: Static testing techniques, Static analysis tools, Unit/Code functional testing, Control flow testing, Code complexity testing, Data flow testing.

UNIT - III

Integration, System and Acceptance Testing: Integration testing approaches, System testing, Scenario Testing, Deployment testing, Non-functional testing techniques,

Acceptance Testing: Acceptance criteria, types, test cases selection and execution.

UNIT - IV

Quality Assurance*:* Quality Planning, Quality plan objectives, Planning process overview, Business Plan and Quality Plan, TQM (Total Quality Management), TQM concepts, Zero defect movement

Quality Standards: Quality Models/Standards, Standards and guidelines, Types of Models, ISO Standards, CMM and CMMI, Six Sigma concepts, Quality Challenge, National Quality Awards.

UNIT - V

Regression testing, Regression test process, Selection of regression tests, Dynamic Slicing, Test Minimization, Tools for regression testing. Test Management and Automation Test Planning, Management, Execution and Reporting, Software Test Automation: Scope of automation, Design & Architecture for automation, Testing tools, Object Oriented Testing.

Text Books:

1. Yogesh Singh, "*Software Testing*", Cambridge University Press, 2011
2. Sagar Naik, "Piyu Tripathy," *Software Testing and Quality Assurance*", Wiley

Reference Books:

1. *Effective methods for Software Testing*, William Perry, Wiley
2. Aditya P. Mathur, "*Foundation of Software Testing*", Pearson Education.
3. Milind Limaye, "*Software Quality Assurance*", McGraw-Hill publication.
4. Paul C. Jorgensen, "*Software Testing: A Craftsman's Approach*", Auerbach Publications, 2008.

Course Code: **CSE353E**	Course Title: **C# and .Net Programming**	Credits: 03 **L - 2 P - 2**

Course Outcomes

- *Understand use of C# basics, Objects and Types, Inheritance.*
- *Develop, implement and creating Applications with C#.*
- *Understand and be able to explain Security in the .NET framework and Deployment in the .NET.*
- *Develop Assemblies and Deployment in .NET, Mobile Application Development etc.*

UNIT – I

Introduction to C#:

Introducing C#, Understanding .NET, Overview of C#, Literals, Variables, Data Types, Operators, Checked and Unchecked Operators, Expressions, Branching, Looping, Methods, Implicit and Explicit casting, Constant, Arrays, Array Class, Array List, String, String Builder, Structure, Enumerations, Boxing and Unboxing.

UNIT - II

Object Oriented Aspects of C#:

Classes, Objects, Constructors and its types, Inheritance, Properties, Indexers, Index overloading, Polymorphism, Sealed class and methods, Interface, Abstract class, Abstract and Interface, operator overloading, Delegates, Events, Errors and Exception, Threading.

UNIT – III

Application Development on .NET:

Building windows application, creating window forms with events and controls, menu creation, inheriting window forms, SDI and MDI application, Dialog Box (Modal and Modeless), accessing data with ADO.NET, Dataset, typed dataset, Data Adapter, updating database using stored procedures, SQL Server with ADO.NET, handling exceptions, validating controls, windows application configuration.

UNIT – IV

Web Based Application Development on .NET:

Programming web application with web forms, ASP.NET introduction, working with XML and .NET, Creating Virtual Directory and Web Application, session management techniques, web.config, web services, passing datasets, returning datasets from web services, handling transaction, handling exceptions, returning exceptions from SQL Server.

UNIT - V

CLR and .NET Framework:

Assemblies, Versoning, Attributes, reflection, viewing meta data, type discovery, reflection on type, marshalling, remoting, security in .NET

Text Books:

1. Herbert Schildt, "*The Complete Reference: C# 4.0*", Tata McGraw Hill, 2012.
2. Christian Nagel, "*Professional C# 6 and .NET Core 1.0*", Wiley, 2016.

Reference Books:

1. Andrew Troelsen, "*Pro C# 2010 and the .NET 4 Platform*", Fifth edition, A Press, 2010.
2. Ian Griffiths, Matthew Adams, Jesse Liberty, "*Programming C# 4.0*", Sixth Edition, O'Reilly, 2010.

Course Code: **CSE354E**	Course Title: **Distributed Computing**	Credits: 03 **L - 3 P - 0**

Course Outcomes

- *Understand the various communications models used by the distributed systems to orchestra their work.*
- *Students will examine how existing systems like Torrent, Chord etc. have applied the concepts of distributed systems in designing large systems.*
- *Be able to design the algorithms for the efficient and scalable work of the small distributed systems.*

UNIT – I

Introduction to Distributed System: Goals, Hardware concepts, Software concepts, and Client-Server model. Examples of distributed systems. Resource sharing and the web, Challenges arising from the construction of distributed systems

System Models: Architectural models Client-server model, Peer-to-peer model, Variations of the above models, Forms of computing Monolithic, Distributed, Parallel, Cooperative

UNIT - II

Networking and Internetworking, Types of networks, Network principles, Internet protocols, Case studies: Ethernet, WIFI, Bluetooth and ATM

UNIT - III

Interposes Communication, The API for the internet protocol, External data representation and marshalling, Client-server communication, Group communication, Remote procedures call, Case study: interposes communication in UNIX

UNIT - IV

Communication: Layered protocols, Remote procedures call, Remote object invocation, Message-oriented communication, Stream-oriented communication, Processes: Threads, Clients, Servers, Code Migration, Software agent, Naming: Naming entities, locating mobile entities, Removing un-referenced entities.

Consistency and Replication: Introduction, Data centric consistency models, Client centric consistency models, Distribution protocols, Consistency protocols.

UNIT - V

Distributing Multimedia: DMS, Advanced Distributed Computing Paradigms: Mobile Agents Security: Introduction, Secure channels, Access control, Security management.

Distributed File System: Sun network file system, CODA files system. Case Study: CORBA, Distributed COM, Comparison of CORBA, DCOM

Text Books:

1. *Distributed Computing: Principles and Applications*, M. L. Liu, Pearson/Addison-Wesley, ISBN: 0-201-79644-9
2. A. Taunenbaum, *Distributed Systems: Principles and Paradigms*
3. G. Coulouris, J. Dollimore, and T. Kindberg, *Distributed Systems: Concepts and Design*, Pearson Education

Course Code: **CSE355E**	Course Title: **Digital Image Processing**	Credits: 03 **L - 3 P - 0**

Course Outcomes

- *Understand the motivation/application areas of digital image processing and the basics involved like image representation, resolution, interpolation, digitization, pixel relationships, etc.*
- *Learn and understand the Image Enhancement techniques in Spatial as well as Frequency Domains*
- *Understand different concepts and approaches of image restoration, redundancies and compression*

UNIT- I:

Introduction, digital image fundamentals, origin of digital image processing, examples of fields that use digital image processing, fundamental steps in image processing, elements of digital image processing systems, image formation model, image representation, image sampling and quantization, resolution, image interpolation, basic relationships between pixels: neighborhood, connectivity, path, connected component labelling algorithm.

UNIT- II:

Distance measures between pixels, linear and non-linear operations, image enhancement in the spatial domain: some basic grey level transformations, histogram: introduction, histogram equalization, histogram specification

Basics of spatial filters, smoothening spatial filters, sharpening spatial filters, filtering in the frequency domain: introduction to Fourier transform and the frequency domain, smoothing and sharpening frequency domain filters

UNIT- III:

Image Compression: fundamentals of compression, coding, spatial and temporal redundancy, lossy and lossless compression, Image compression models, basic compression methods.

Morphology: introduction, some basic Morphological operators and algorithms

UNIT- IV:

Image Segmentation: introduction, detection of discontinuities, edge linking and boundary detection, region-oriented segmentation, motion-based segmentation.

Image Restoration and Reconstruction: introduction and basic concepts.

UNIT- V:

Representation: introduction, various approaches. Description: introduction, boundary descriptors, regional descriptors,

Text Books:

1. Rafael C. Gonzalez & Richard E. Woods, "*Digital Image Processing*", 3rd edition, Pearson, 2002.
2. A.K. Jain, "*Fundamental of Digital Image Processing*", PHI, 1989.

Reference Books:

1. Bernd Jahne, "*Digital Image Processing*", 5th Ed., Springer, 2002.
2. William Ks Pratt, "*Digital Image Processing: Piks Inside*", John Wiley & Sons, 2001.

Course Code: **CSE356E**	Course Title: **Programming in R**	Credits: 03 **L - 2 P - 2**

Course Outcomes

- *Understand the fundamental syntax of R through readings, practice exercises, demonstrations, and writing R code.*
- *Import a variety of data formats into R using RStudio.*
- *Perform basic analytical operations on various data sets.*
- *Analyze a data set in R and present findings using the appropriate R packages.*

UNIT - I

What is R, Basic Features of R, Design of the R System, Limitations of R, R Resources, R Installation; Entering Input, Evaluation, R Objects, Numbers, Attributes, Creating Vectors, Mixing Objects, Explicit Coercion, Matrices, Lists, Factors, Missing Values, Data Frames, Names.

UNIT - II

Getting Data In and Out of R: Reading and Writing Data, Reading Data Files with *read.table()*, Reading in Larger Datasets with *read.table()*, Memory Requirements for R Objects; Using the *readr* Package; Using Textual and Binary Formats for Storing Data: Using *dput()* and *dump()*, Binary Formats; Interfaces to the Outside World: File Connections, Reading Lines of a Text File, Reading From a URL Connection; Subsetting R Objects: Subsetting Vector, Matrix, Lists, Nested Elements of a List, Extracting Multiple Elements of a List, Partial Matching, Removing NA Values; Vectorized Matrix Operations; Date and Time in R, Operations on Date and Time.

UNIT - III

Managing Data Frames with the dplyr package: Data Frames, The *dplyr* Package, *dplyr* Grammar, Installing the *dplyr* package, *select ()*, *filter ()*, *arrange()*, *rename()*, *mutate()*, *group_by()*, *%>%*; Control Structures: *if-else*, *for* Loops, Nested *for* loops, *while* Loops, *repeat* Loops, *next*, *break*;

UNIT - IV

Functions: Functions in R, Your First Function, Argument Matching, Lazy Evaluation, The ... Argument, Arguments Coming After the ... Argument; Scoping Rules of R: A Diversion on Binding Values to Symbol, Scoping Rules, Lexical Scoping: Why Does It Matter, Lexical vs. Dynamic Scoping, Application: Optimization, Plotting the Likelihood; Coding Standards for R; Loop Functions: Looping on the Command Line, *lapply()*, *sapply()*, *split()*, Splitting a Data Frame, *tapply*, *apply()*, Col/Row Sums and Means, Other Ways to Apply, *mapply()*, Vectorizing a Function;

UNIT - V

Debugging: Figuring Out What's Wrong, Debugging Tools in R, Using *traceback()*, *debug()*, *recover()*; Profiling R Code: Using *system.time()*, Timing Longer Expressions, The R Profiler, Using *summaryRprof()*; Simulation: Generating Random Numbers, Setting the random number seed, Simulating a Linear Model, Random Sampling; Plotting: Five-Number Summary, Boxplots, Barplots, Histograms, Scatterplots, Density plots, Graphs. Data Analysis Case Study.

Text Book:

1) Roger D. Peng, "*R Programming for Data Science*", Leanpub, 2015.

Reference Books:

1. Roger D. Peng, "*Exploratory Data Analysis with R*", Leanpub, 2015.
2. Hadley Wickham and Garrett Grolemund, "*R for Data Science*", O'Reilly Media, 2017.

Course Code: **CSE356E**	Course Title: **Soft Computing**	Credits: 03 **L - 3 P - 0**

Course Outcomes

- *Ability to analyze and appreciate the applications which can use fuzzy logic.*
- *Ability to design inference systems.*
- *Ability to understand the difference between learning and programming and explore practical applications of Neural Networks (NN).*
- *Ability to appreciate the importance of optimizations and its use in computer engineering fields and other domains.*

UNIT - I

Neural Networks:
History, overview of biological Neuro-system, Mathematical Models of Neurons, ANN architecture, Learning rules, Learning Paradigms-Supervised, unsupervised and reinforcement Learning, ANN training Algorithms perceptions, Training rules, Delta, Back Propagation Algorithm, Multilayer Perceptron Model, Hopfield Networks, Associative Memories, Applications of Artificial Neural Networks.

UNIT - II

Fuzzy Logic:
Introduction to Fuzzy Logic, Classical and Fuzzy Sets: Overview of Classical Sets, Membership Function, Fuzzy rule generation. Operations on Fuzzy Sets: Compliment, Intersections, Unions, Combinations of Operations, Aggregation, Operations.

UNIT - III

Fuzzy Arithmetic:
Fuzzy Numbers, Linguistic Variables, Arithmetic Operations on Intervals & Numbers, Lattice of Fuzzy Numbers, Fuzzy Equations. Fuzzy Logic: Classical Logic, Multivalued Logics, Fuzzy Propositions, Fuzzy Qualifiers, Fuzzy Decision Making, Fuzzy Control Systems.

UNIT - IV

Introduction of Neuro-Fuzzy Systems:
Neuro-Fuzzy Hybrid Systems – Genetic Neuro Hybrid Systems – Genetic Fuzzy Hybrid and Fuzzy Genetic Hybrid Systems.

UNIT – V

Genetic Algorithm:
Concept of "Genetics" and "Evolution" and its application to probabilistic search techniques. Basic GA framework and different GA architectures. GA operators: Encoding, Crossover, Selection, Mutation, etc. Solving single-objective optimization problems using GAs.

Text Books:

1. J.S.R.Jang, C.T. Sun And E.Mizutani, "*Neuro-Fuzzy And Soft Computing*", PHI / Pearson Education 2004.
2. S.N.Sivanandam & S.N.Deepa, "*Principles of Soft Computing*", Wiley India Pvt Ltd, 2011.
3. F. O. Karray and C. de Silva, "*Soft computing and Intelligent System Design*", Pearson, 2009.

Reference Books:

- S.Rajasekaran & G.A.Vijayalakshmi Pai, "*Neural Networks, Fuzzy Logic And Genetic Algorithm: Synthesis & Applications*", Prentice Hall Of India Pvt. Ltd., 2006.
- "*Neural Networks – A Comprehensive Foundations*", Prentice-Hall International, New Jersey, 1999.
- Freeman J.A. & D.M. Skapura,"*Neural Networks: Algorithms, Applications and Programming Techniques*", Addison Wesley, Reading, Mass, (1992).

B-Tech Computer Science & Engineering
Syllabus for Semester-VII

Course Code	Course Title	Hours Per Week		Credit
		L	P	
CSE401C	Compiler Design	3	0	3
CSE402C	Network Security	3	0	3
	Elective –II	3	0	3
	Elective –III	3	0	3
CSE403C	Compiler Design Lab	0	2	1
CSE404C	Network Security Lab	0	2	1
CSE405C	Project II	0	6	3
CSE406C	Seminar	0	0	1
Total Credits		**12**	**10**	**18**

Course Code: **CSE401C**	Course Title: **Compiler Design**	Credits: 03 **L - 3 P - 0**

Course Outcomes

- *Understand and explain the main techniques and algorithms used in compilers.*
- *To learn basic data structures used in compiler construction such as abstract syntax trees, symbol tables, three address code and stack machines.*
- *To understand the principles and techniques used to perform translation and the fundamental concepts of translator construction.*
- *Acquired the skills to understand, develop, and analyze recognizers for programming languages.*

UNIT - I

Introduction to compilers, Phases of Compiler, Compiler construction tools, Classification of grammars, Context free grammars.

UNIT - II

Scanners, Top down parsing, LL grammars, Bottom up parsing, Polish expression Operator, Precedence grammar, IR grammars.

UNIT - III

Comparison of parsing methods, Error handling. Symbol table handling techniques, Organization for non-block and block structured Languages

UNIT - IV

Run time storage administration, Static and dynamic allocation, Intermediate forms of source program, Polish N-tuple and syntax trees, Semantic analysis and code generation.

UNIT - V

Code optimization, Folding, redundant sub-expression evaluation, Optimization within Iterative loops.

TEXT BOOKS:

1. Aho, Ullman & Ravi Sethi, "*Principles of Compiler Design*", Pearson Education

REFERENCE BOOKS:

- Tremblay, et. al., "*The Theory and Practice of Compiler Writing*", McGraw Hill, New York
- Holub, "*Compiler Design in C*", PHI
- Andrew L. Appel, "*Modern Compiler Implementation in C*", Delhi Foundation Books
- Dick Grune et. Al., "*Modern Compiler Design*", John Wiley and Sons

Course Code: **CSE402C**	Course Title: ***Network Security***	Credits: 03 **L - 3 P - 0**

Course Outcomes

- *Understand the fundamentals of cryptography*
- *Understand how the key distribution is done between the communicating parties*
- *Acquire knowledge on standard algorithms used to provide confidentiality, Integrity and authenticity.*
- *Understand the usage of encryption techniques to secure the data in transit across the data networks Configure the firewall system to secure the networks from external security threats.*

Unit - I

Introduction to cryptography, Symmetric and Asymmetric Ciphers, Data Encryption Standard (DES), Triple DES, Design and analysis, AES, RC4.

Unit - II

Introduction to Public Key Cryptography, Principles, Public-Key Cryptography Algorithms (RSA), Approaches to Message: Authentication, Secure Hash Functions, Message Authentication Codes (MAC). Diffie-Hellman Key Exchange.

Unit - III

Modular Inverse, Extended Euclid Algorithm, Fermat's Little Theorem, Euler Phi-Function, Euler's theorem**.**

Unit - IV

Web Security Considerations, Secure Socket Layer / Transport Layer Security (SSL/TLS), HTTPS, Secure Shell (SSH), Good Privacy (PGP),

Unit - V

Wireless security, DNS security, Smartcards/Biometrics.

Introduction to Quantum Cryptography, Blockchain, Bitcoin and Cryptocurrency.

Book Recommended

1. William Stallings, "*Cryptography and Network Security – Principles and Practices*".
2. Behrouz A. Forouzan, "*Cryptography and Network Security*".
3. Charlie Kaufman, Radia Perlman and Mike Speciner, "*Network Security: Private Communication in a Public World*".

REFERENCES

- Bruce Schneier, "*Applied Cryptography*", John Wiley & Sons Inc, 2001.
- Atul Kahate, "*Cryptography and Network Security*", Tata McGraw Hill, 2003.
- Charles B. Pfleeger, Shari Lawrence Pfleeger, "*Security in Computing*", Third Edition, Pearson Education, 2003.
- *Networking Essentials*, by William. S.Stallings

Course Code: **CSE403C**	Course Title: **Compiler Design Lab**	Credits: 01 **L - 0 P - 2**

Course Outcomes

- *To acquaint students with software tools and techniques which are applicable both to compilers and the implementation of system utility routines, command interpreters, etc.*
- *Use compiler construction tools, such as parser generators, to build a simple compiler.*
- *Demonstrate a working understanding of the process of lexical analysis, parsing and other compiler design aspects.*
- *To implement Lexical Analyzer using Lex tool & Syntax Analyzer or parser using YACC Tool.*
- *To implement code optimization techniques.*

Lab Details:

1. Design NFA/DFA to recognize an identifier.
2. Design a Lexical analyzer for the given language.
3. Implement the lexical analyzer using JLex, flex or lex or other lexical analyzer generating tools.
4. Design Predictive parser for the given language
5. Design LALR bottom up parser for the given language.

Course Code: **CSE404C**	Course Title: **Network Security Lab**	Credits: 01 **L - 0 P – 2**

Course Outcomes

- *Analyze the vulnerabilities in any computing system.*
- *Implement basic ciphers and cryptographic algorithms.*
- *Implement authentication Schemes E.g. Digital Signature*
- *Implement various networking protocols and to protect the network from threats.*

LIST OF EXPERIMENTS:

1. Implement the following SUBSTITUTION & TRANSPOSITION TECHNIQUES concepts:
    1. Caesar Cipher
    2. Playfair Cipher
    3. Hill Cipher
    4. Vigenere Cipher
    5. Rail fence – row & Column Transformation
2. Implement the following algorithms
    1. DES
    2. RSA Algorithm
    3. Diffiee-Hellman
    4. MD5
    5. SHA-1
3. Implement the SIGNATURE SCHEME – Digital Signature Standard
4. Demonstrate how to provide secure data storage, secure data transmission and for creating digital signatures (GnuPG).
5. Setup a honey pot and monitor the honeypot on network (KF Sensor)
6. Installation of rootkits and study about the variety of options
7. Perform wireless audit on an access point or a router and decrypt WEP and WPA. (Net Stumbler)

Course Code: **CSE405C**	Course Title: **Project II**	Credits: 03 **L - 0 P – 6**

Description

During 7th semester, students have to take Project – II (Minor – Project) of 3 credits. In Project – II, students identify a problem in their field of interest and also try to do at least partial implementation of their stated problem. This would help them to extend their Project – II to Project – III (Major – Project).

Course Code: **CSE406C**	Course Title: **Seminar**	Credits: 01 **L - 0 P – 0**

Description

In 7th semester students have to take one credit course titled 'Seminar'. In this course they have to appear for Power Point Presentation on a pre-approved topic from the department. The topic of the seminar should be chosen by keeping in view the **"current technological trends and advancements"** in computer science and engineering. A departmental committee will evaluate the individual students' performance on the basis of their presentation and knowledge about the topic chosen for seminar.

B. Tech. Computer Science and Engineering
Syllabus of Discipline Centric Electives for
Semester Seventh

Course Code	Course Title	Hours Per Week		Credits	Pre-requisite	Semester
		L	P			
CSE401E	Advanced JavaScript	2	2	03	Introduction to Core Java Script	7th
CSE402E	Python Programmi	2	2	03	Nill	7h
CSE403E	Machine Learning	3	0	03	Computer Programming	7th
CSE404E	Multimedia Technology	3	0	03	Nill	7th
CSE405E	Web Technologies	2	2	03	I&WD and JAVA	7th
CSE406E	Advanced Java	2	2	03	Java	7th
CSE407E	Data Mining	3	0	03	DBMS	7th
CSE408E	Robotics	3	0	03	Embedded system	7th
CSE409E	Internet of Things	3	0	03	Computer Networks/ Embedded Systems	7th
CSE410E	Agile Software Development	3	0	03	Software Engineering	7th
CSE411E	GO Language	2	2	03	Nill	7th
CSE412E	Modelling and Simulation	3	0	03	Computer Programming	7th

Course Code: **CSE401E**	Course Title: **Advanced JavaScript**	Credits: 03 **L - 2 P – 2**

Course Outcomes

- *Write applications that manipulate the Document Object Model to fetch and display information using jQuery.*
- *Create anonymous functions and closures, and use them to store and access local data.*
- *Create event listeners and callbacks to respond to user-interface and network events.*
- *Test and debug JavaScript web applications.*

Unit - 1:

Introduction to Object Oriented Approach Object Oriented Approach vs Functional Approach, Setting Up Development Environment NodeJS, Live Server. Installing required software's (Visual Studio Code/Atom /). Objects Literals, Constructors, Enumerating Properties, Abstraction, Private Properties and Methods, Getters and Setters.

Unit 2:

Prototypical Inheritance Creating Prototypical Inheritance, Calling the Super Construction, Method Overriding, Polymorphism.

Unit 3:

ES6 Classes Introduction to ES6 Classes, Hoisting, Static Methods, The this Keyword, Private Members, Getters and Setters, Inheritance, Method Riding

Unit 4:

ES6 Modules and Asynchronous JavaScript Asynchronous vs Synchronous Execution, Closures, allbacks, Promises, Promise Chaning, Fetch API, Async Await. Modules, Using Modules, ES6 Modules, Introduction to Babel and Webpack .

Unit 5:

Introduction to NodeJS What is NodeJS, NodeJS Architecture, Modules in Node (OS, FileSystem, HTTP), Node Package Manager , Installing Packages using npm , Introduction to Express , Serving Files using Express .

References : https://www.w3schools.com/js/default.asp (w3 Schools)
Mozilla Developer Network (MDN)
https://www.w3schools.com/nodejs/ (w3 Schools NodeJS)

Course Code: **CSE402E**	Course Title: **Python Programming**	Credits: 03 **L - 2 P – 2**

Course Outcomes

- *design and program Python applications*
- *use lists, tuples, and dictionaries in Python programs.*
- *use indexing and slicing to access data in Python programs*
- *write loops and decision statements in Python*
- *build and package Python modules for reusability.*
- *design object-oriented programs with Python classes.*

UNIT - I: Introduction to Python and Data types

Installation and Working with Python, Understanding Python variables, Python basic Operators, Understanding python blocks. Declaring and using Numeric data types: int, float, complex, using string data type and string operations, defining list and list slicing, Use of Tuple data type.

UNIT – II: Python Flow Control

Conditional blocks using if, else and elif, Simple for loops in python, For loop using ranges, string, list and dictionaries, use of while loops in python, Loop manipulation using pass, continue, break and else, Programming using Python conditional and loops block.

UNIT – III: Python Functions, Modules and Packages

Organizing python codes using functions, importing own module as well as external modules, Understanding Packages, Powerful Lambda function in python, Programming using functions, modules and external packages

UNIT – VI: Python string, list, dictionary manipulation and file operations

Building blocks of python programs, understanding string in build methods, List manipulation using in build methods, Dictionary manipulation, Programming using string, list and dictionary in build functions, python file operations.

UNIT – V: Python Object Oriented Programming

Concept of class, object and instances, Inheritance, overlapping and overloading operators, Programming using Oops support. Introduction to libraries viz. numpy, pyplot and pandas.

RECOMMENDED TEXTBOOKS:

1. Mark Pilgrim, — "Dive into Python 3", Apress, 2009.
2. Allen Downey, Jeffrey Elkner, Chris Meyers, — "*How to Think Like a Computer Scientist – Learning with Python*", Green Tea Press,2002.

REFERENCES

1. John V. Guttag, — "*Introduction to Computation and Programming using Python*", Prentice Hall of India, 2014.
2. Mark Lutz, — "*Learning Python: Powerful Object-Oriented Programming"*, Fifth Edition, O'Reilly, Shroff Publishers and Distributors, 2013.

Course Code: **CSE403E**	Course Title: **Machine Learning**	Credits: 03 **L - 3 P – 0**

Course Outcomes

- *Identify and pose classification and regression problems.*
- *Implement various machine learning algorithms and get a know how of many in vouge ML libraries.*
- *Compare and study the performance of various algorithms*
- *Work on a machine learning project from data organization to selection of proper ML algorithm for problem solving.*

Unit I: Introduction

Introduction to Machine Learning. Types of Learning. Generative and Discriminative algorithms.
Review of basic concepts of Probability Theory, Linear Algebra and Optimizations.

Unit II: Supervised Learning

Linear Regression, Logistic Regression, SoftMax Regression Perceptron. Bayes Classifier and Naïve Bayes Classifier.

Unit III Structured Machine Learning:

Model selection and feature selection Bias-variance trade-off, overfitting and underfitting. training, validation and test split. Cross validation. Regularization
Introduction to machine learning libraries – scikit learn

Unit IV: More Supervised Learning

Decision Trees, Support Vector Machines, Neural Networks, Backpropagation algorithms and Gradient Descent

Unit V: Unsupervised Learning

Clustering. K-means. Expectation Maximization. Mixture of Gaussians. PCA (Principal components analysis). ICA (Independent components analysis).

Books Recommended:

1. *Machine Learning. Tom Mitchell*. First Edition, McGraw- Hill, 1997.
2. *Introduction to Machine Learning*, Edition 2, by Ethem Alpaydin
3. *The Elements of Statistical Learning Data Mining, Inference, and Prediction,* Trevor Hastie, Robert Tibshirani, Jerome Friedman, Spinger.

Course Code: **CSE404E**	Course Title: **Multimedia Technology**	Credits: 03 **L - 3 P – 0**

Course Outcomes

- *Be familiar with multimedia data types and the conversion between analogue and digital forms.*
- *Have gained experience in the use of multimedia systems and the ability to manipulate multimedia data programmatically.*
- *Have gained an understanding of the issues that arise when multimedia communication is attempted across the Internet.*
- *Be familiar with various performance issues related to multimedia system design.*

Unit - I

Introduction: What is Multimedia, Multimedia Information, Multimedia Objects, Multimedia in business and work. Convergence of Computer, Communication and Entertainment products.
Stages of Multimedia Projects: Multimedia hardware, Memory & storage devices, Communication devices, Multimedia software's, presentation tools, tools for object generations, video, sound, image capturing, authoring tools, card and page-based authoring tools.

Unit - II

Multimedia Building Blocks: Text, Sound MIDI, Digital Audio, audio file formats, MIDI under windows environment Audio & Video Capture.

Unit - III

Data Compression: Huffman Coding, Shannon Fano Algorithm, Huffman Algorithms, Adaptive Coding, Arithmetic Coding Higher Order Modelling. Finite Context Modelling, Dictionary based Compression, Sliding Window Compression, LZ77, LZW compression, Compression, Compression ratio loss less & lossy compression.

Unit - IV

Speech Compression & Synthesis: Digital Audio concepts, Sampling Variables, Loss less compression of sound, loss compression & silence compression.

Unit - V

Images: Multiple monitors, bitmaps, Vector drawing, lossy graphic compression, image file format, animations Images standards, JPEG Compression, Zig Zag Coding, Multimedia Database. Content based retrieval for text and images,
Video: Video representation, Colours, Video Compression, MPEG standards, MHEG Standard Video Streaming on net, Video Conferencing, Multimedia Broadcast Services, Indexing and retrieval of Video Database, recent development in Multimedia.

Text Books:

1. Tay Vaughan "*Multimedia, Making it Work*" Osborne McGraw Hill.
2. Buford "Multimedia Systems" Addison Wesley.

Reference Books:

- Mark Nelson "*Data Compression Book*" BPB.
- Sleinreitz "*Multimedia System*" Addison Wesley.
- Agrawal & Tiwari "*Multimedia Systems*" Excel.
- David Hillman "*Multimedia technology and Applications*" Galgotia Publications.
- Rosch "*Multimedia Bible*" Sams Publishing.
- James E Skuman "*Multimedia in Action*" Vikas.

Course Code: **CSE405E**	Course Title: **Web Technologies**	Credits: 03 **L - 2 P - 2**

Course Outcomes

- *Have knowledge about internet history and the skill required to effectively utilize the internet resources*
- *Able to apply the basics to design a web page and identify its elements and attributes.*
- *Know all about web search and e-learning and online file storage*
- *Learning the basics about web browsing by HTTP and URL*
- *Able to identify and defend against the risk in system.*

Unit – I

XML: Document type definition, XML Schemas, Document Object model, Presenting XML, Using XML Processors: DOM and SAX, Overview of, XPath, XQuery, and XSLT. Ajax Improving web page performance using Ajax, Programming in Ajax.

Unit – II

Objects in java script: Such as Array, date, math, string, accessing page elements using DOM (Document object model). Introduction to J2EE architecture: Single Tier Two Tier Three Tier N Tier.
J2EE Containers: Container Types, Container Services.

Unit – III

JAVA servlets: Introduction to Web Programming, Advantages of Servlets, Servlet Lifecycle, Request Dispatching, Session Tracking.

Unit – IV

JAVA SERVER PAGES (JSP) & JSTL: JSP Architecture, JSP Objects, Custom Tags, JSP Elements, Using tags of JSTL, expression languages

Unit – V

Overview of MVC, Implementing Model-View-Controller, The Struts Model, Deploying Action Servlet, Action Form, Action Mapping and Action Errors, Relational Data.

Books Recommended

1. Deitel and Deitel, Goldberg, "I*nternet and World Wide Web – How to Program*", Pearson Education Asia, 2001.
2. *Professional Jakarta Struts* – James Goodwill, Richard Hightower Wrox publication
3. Mathew Eernisse, "*Build Your Own AJAX Web Applications*", SitePoint, 2006.
4. *JavaScript: The Definitive Guide, Fourth Edition* – By David Flanagan O'Reilly Media
5. *Professional Java Server Programming J2EE edition* Volume I, II, Wrox publications
6. *Beginning XML'* by David Hunter and David Gibbons, Wrox publication
7. "*Using HTML 4, XML and JAVA*", Eric Ladd, Jim O' Donnel, Prentice Hall of India.

Course Code: **CSE406E**	Course Title: **Advanced Java**	Credits: 03 **L - 2 P - 2**

Course Outcomes:

- *Identify advance concepts of java programming with database connectivity.*
- *Develop client/server applications and TCP/IP socket programming.*
- *Able to implement the concepts of Hibernate, XML& EJB for building enterprise applications.*
- *Design and develop platform independent applications using a variety of component-based frameworks.*

UNIT - I

Java Platform, Enterprise Edition: Java EE Platform Overview, Java EE Platform – Distributed Multi-Tiered Applications, Java EE – Web & Business Components, Java EE Containers – services & types, Java EE Application Assembly & Deployment – Packaging Applications, Java EE modules, Getting Started with Web Applications, Model View Controller (MVC)2 Architecture & Packaging EJB Module

Application Deployment: Web application development and deployment Steps, Configuring Web application – Web application deployment descriptor (web.xml file), Web Application Archive (*.WAR file) – *.WAR directory structure, Building & Deploying Applications, Ant build tool.

UNIT - II

Java Database Connectivity: JDBC Product, Types of Drivers, Two-Tier Client/Server Model, Three-Tier Client/Server Model, Basic Steps of JDBC, Creating and Executing SQL Statement, The Result Set Object, Working with Database Metadata.

UNIT - III

Enterprise Java Beans: Types of Enterprise Java beans, Session Bean & Entity Bean, Features of Session Bean, Life-cycle of Stateful Session Bean, Features of Entity Bean, Life-cycle of Entity Bean, Container managed Transactions & Bean-managed Transactions, Implementing a container-managed Entity Bean

UNIT - IV

Struts: - Introduction to the Apache Struts, MVC Architecture, Struts Architecture, How Struts Works? Introduction to the Struts Controller, Introduction to the Struts Action Class, Using Struts Action From Class, Using Struts HTML Tags, Introduction to Struts Validator Framework, Client Side Address Validation in Struts, Custom Validators Example, Developing Application with Struts Tiles

UNIT- V

Introduction to Hibernate: ORM Overview, Hibernate Environment, Hibernate Architecture & API, Hibernate Configuration, Hibernate Sessions, Persistent Class & Mapping Files, Building Hibernate application, Hibernate Query Language (HQL), Hibernate O/R Mappings – Collection & Association Mappings- Many-to-One, One-to-One, One-to-Many, Many-to-Many, Implementing Hibernate in Java Web Applications

References:

- *Professional Java Server Programming J2EE edition*, Volume I, II Wrox publications
- *Professional Jakarta Struts*, James Goodwill, Richard Hightower Wrox publication
- *Java for web with Servlets, JSP and EJB*, by Budi Kurniwan Techmedia publications

Course Code: **CSE407E**	Course Title: **Data Mining**	Credits: 03 **L - 3 P - 0**

Course Outcomes

- *At the end of the course, students:*
- *Can define what data mining and data warehousing is and what it can be used for.*
- *Can determine the different steps followed in Data mining and pre-processing for data mining.*
- *Can describe and apply at least two of the algorithms used for generating frequent patterns and association rules for data mining.*
- *Can apply at least two of the Classification methods for data mining.*

UNIT - I

Introduction: What is data mining, Kinds of data that can be mined, Kinds of patterns that can be mined; **Knowing data:** Types of attributes, Basic statistical descriptions of data (mean, median, mode, range, quartiles, variance, standard deviation), Measuring data similarity and dissimilarity (for each attribute type), cosine similarity; **Data pre-processing:** Data cleaning, Data integration, Data reduction, Data transformation, Data discretization.

UNIT - II

Data warehouse: What is a data warehouse, Operational database system verses data warehouse, Data warehousing architecture; **Data warehouse modelling:** Data cube, Schemas for multidimensional data models (star, snowflake, fact constellation), Online analytical processing operations (roll-up, drill-down, slice & dice, pivot, etc.), OLAP server architectures, From OLAP to multidimensional data mining.

UNIT - III

Frequent pattern mining: Basic concepts, Market basket analysis, Frequent itemset mining algorithms (Apriori algorithm, FP-growth algorithm, Eclat algorithm), Mining closed and maximal patterns, Association rule mining, Correlation analysis.

UNIT - IV

Classification: What is classification, Decision tree induction, Attribute selection measures, Tree pruning, Bayes' theorem, Naive Bayesian classification, Rule-based classification, Metrics for evaluating classifier performance, Ensemble methods, Classification by Backpropagation (Neural network-based classification), Support vector machines (SVMs), K-nearest-neighbor (KNN) classifier.

UNIT - V

Clustering: What is clustering; Partitioning methods for clustering: K-means clustering, K-medoids clustering; Hierarchical methods: AGNES, DIANA, BIRCH; Density-based method: DBSCAN; Probabilistic model-based method: Expectation-Maximization algorithm.

Text Book:

1. Jiawei Han, Micheline Kamber and Jian Pei, "Data Mining: Concepts and Techniques", Morgan Kaufmann Publishers, Elsevier, Third Edition, 2011.

Reference Books:

- Pang-Ning Tan, Michael Steinbach and Vipin Kumar, "Introduction to Data Mining", Addison-Wesley, Pearson, 2005.
- Mohammed J. Zaki, Wagner Meira Jr., "Data Mining and Analysis: Fundamental Concepts and Algorithms", Cambridge University Press, 2014.

Course Code: **CSE408E**	Course Title: **Robotics**	Credits: 03 **L - 3 P - 0**

Course Outcomes

- *Understand the key components of a robot.*
- *Ability to identify the sensory needs for a robot.*
- *Ability to design a good interface for the robot.*
- *Ability to evaluate the computational needs of a robot and selecting a proper system for designing a robot.*
- *Ability to develop a robotic solution for the given environment.*

Unit - I: Basics of Robotics and Linear Algebra

Introduction, Spatial descriptions and transformations, manipulator kinematics, Robot programming language and systems, some inverse kinematic formulas.

Unit – II: Microcontroller in Robots

Support components - Memory and device programming – Interrupts - Built in peripherals - Interfacing the controller to robots.

Unit – III: Development Process

Managing Design Process: Introduction – Organizational Design to Support Usability – Four Pillars of Design – Development Methodologies – Ethnographic Observation – Participatory Design – Scenario Development – Evaluating Interface Design: Expert Reviews – Usability Testing and Laboratories – Survey Instruments – Acceptance Test – Evaluation During Active Use – Controlled Psychologically Oriented Experiments.

Unit – IV: Manipulation and Virtual Environments

Introduction-Examples of Direct Manipulation Systems –Discussion of Direct Manipulation-3D Interfaces – Teleoperation – Virtual Augmented Reality – Menu Selection, Form Fill-in, and Dialog Boxes: Task-Related Menu organization –Single Menus – Combinations of Multiple Menus – Form Fill-in, Dialog Boxes, and Alternatives – Command and Natural Languages: Command –Organization Functionality, Strategies, and Structure – Naming and Abbreviations – Natural Language in Computing.

Unit – V: Interaction Devices

Introduction – Keyboards and Keypads – Pointing Devices – Speech and Auditory Interfaces – Small and Large Displays – Collaboration and Social Media Participation: Goals of Collaboration and Participation – Asynchronous Distributed Interfaces – Synchronous Distributed Interfaces – Face to Face Interfaces -Balancing Function and Fashion: Error Messages – Nonanthropomorphic Design –Display Design –Web Page Design – Window Design – Color.

References

- Michael slater, "*Microprocessor – based design: A comprehensive Guide to Effective Hardware* Design", Prentice Hall, 1989.
- Myke Predko, "*Programming and customizing the 8051 microcontrollers*", Tata McGraw Hill, New Delhi, 2000.
- Kenneth J. Ayala, "*The 8051 micro-controller architecture, programming and applications*", Penram International publishers, Mumbai, 1996.
- Murphy Robin R, "*Introduction to AI Robotics*", MIT Press, 2000.
- Siegwart R and Nourbakhsh I.R, "*Introduction to Autonomous mobile Robots*", Prentice Hall India, 2005.

Course Code: **CSE409E**	Course Title: ***Internet of Things***	Credits: 03 **L - 3 P - 0**

Course Outcomes

- *Defining the role of Internet of Things in ICT with its applications and possible future trends.*
- *Understand State of the Art - IoT Architecture and to classify Real World IoT Design Constraints, Industrial Automation in IoT.*
- *Demonstrating the various network protocols used in IoT and its enabling technologies such as WiFi, 6LoWPAN, bluetooth and ZigBee.*
- *Understand and explain the role of big data, cloud computing and data analytics in a typical IoT system.*

Unit - I

Sensing: Sensors and Transducers, Sensor Classes, Sensor Types, Sensor node architecture, Actuation: Actuator Basics, Actuator Types, Wireless Sensor Networks (WSN), Mobile Ad-hoc Networks (MANET), Introduction to IoT, Definition, Characteristics

Unit - II

IoT architecture: 3-layer, 4-layer, 5 layer, Protocol Classification, basics of IoT networking: RFID, NFC, Zwave, IEEE 802.15.4, ZigBee, 6LoWPAN, MQTT, CoAP, XMPP, AMQP,

Unit - III

IoT Elements, Machine to Machine Communication, Interoperability in IoT, Software Defined Networking (SDN), Cloud Computing, Fog Computing, Big data analytics

Unit - IV

IoT Challenges: Availability, Mobility, Management, Scalability, Interoperability, Security, Other challenges

Unit - V

Smart city, Intelligent Transportation Systems (ITS), Smart Grid, Industrial Internet of Things, Smart healthcare, Surveillance, Other IoT applications

Reference Books:

1. Vijay Madisetti and Arshdeep Bahga, "*Internet of Things (A Hands-on-Approach)",* 1st Edition, VPT, 2014
2. Francis daCosta, "*Rethinking the Internet of Things: A Scalable Approach to Connecting Everything*", 1st Edition, Apress Publications, 2013
3. Cuno Pfister, *Getting Started with the Internet of Things*, O"Reilly Media, 2011, ISBN: 978-1-44939357-1

Course Code: **CSE410E**	Course Title: ***Agile Software Development***	Credits: 03 **L - 3 P - 0**

Course Outcomes

- *Understand the fundamental principles and practices associated with each of the agile development methods*
- *Use agile scrum framework*
- *Execute Agile testing*
- *Practices agile software design and development practices*
- *To perform in-depth explorations into aspects of agile development that are particularly relevant to state-of-art practices*

UNIT - I

Fundamentals of Agile: The Genesis of Agile, Introduction and background, Agile Manifesto and Principles, Overview of Scrum, Extreme Programming, Feature Driven development, Lean Software Development, Agile project management, Design and development practices in Agile projects, Test Driven Development, Continuous Integration, Refactoring, Pair Programming, Simple Design, User Stories, Agile Testing, Agile Tools

UNIT - II

Agile Scrum Framework: Introduction to Scrum, Project phases, Agile Estimation, Planning game, Product backlog, Sprint backlog, Iteration planning, User story definition, Characteristics and content of user stories, Acceptance tests and Verifying stories, Project velocity, Burn down chart, Sprint planning and retrospective, Daily scrum, Scrum roles – Product Owner, Scrum Master, Scrum Team, Scrum case study, Tools for Agile project management.

UNIT - III

Agile Testing: The Agile lifecycle and its impact on testing, Test-Driven Development (TDD), xUnit framework and tools for TDD, Testing user stories - acceptance tests and scenarios, Planning and managing testing cycle, Exploratory testing, Risk based testing, Regression tests, Test Automation, Tools to support the Agile tester.

UNIT - IV

Agile Software Design and Development: Agile design practices, Role of design Principles including Single Responsibility Principle, Open Closed Principle, Liskov Substitution Principle, Interface Segregation Principles, Dependency Inversion Principle in Agile Design, Need and significance of Refactoring, Refactoring Techniques, Continuous Integration, Automated build tools, Version control.

UNIT - V

Industry Trends, Market scenario and adoption of Agile, Agile ALM, Roles in an Agile project, Agile applicability, Agile in Distributed teams, Business benefits, Challenges in Agile, Risks and Mitigation, Agile projects on Cloud, Balancing Agility with Discipline, Agile rapid development technologies.

References:

1. *Agile Software Development with Scrum* By Ken Schawber, Mike Beedle Publisher: Pearson
2. *Agile Software Development, Principles, Patterns and Practices*, By Robert C. Martin Publisher: Prentice Hall
3. *Agile Testing: A Practical Guide for Testers and Agile Teams*, By Lisa Crispin, Janet Gregory Publisher: Addison Wesley
4. *Agile Software Development: The Cooperative Game, By* Alistair Cockburn Publisher: Addison Wesley
5. *User Stories Applied: For Agile* Software, By Mike Cohn

Course Code: **CSE411E**	Course Title: **GO Language**	Credits: 03 **L - 2 P - 2**

Course Outcomes

- *learn Go programming language fundamentals.*
- *apply GoRoutines and channels to build massive parallel systems.*
- *Design and develop the projects using Go runtime.*
- *use Interfaces to simplify complex programs.*
- *build real-time apps using Golang*

UNIT - I

Go overview: features of Go programming, Go environment setup: text editor, Go compilers, Go program structure, Go basic syntax: tokens, line separator, comments, identifiers, keywords

Go data types: integers, float, numeric, variables: static type declaration, dynamic type declaration, mixed variable declaration, lvalue and rbalue, constants

UNIT - II

Go operators: Arithmetic operators, relational operators, logical operators, bitwise operators, assignment operators, miscellaneous operators, operators precedence in Go

Go decision making: if statement, else if statement, nested if statement, switch statement, select statement

UNIT - III

Go loops: for loop, nested loop, loop control statements: break, continue, goto, infinite loop

Go functions: defining a function, parameters, return type, calling a function, call by value, call by reference, Go scope rules: local variables, global variables, formal parameters, initializing local and global variables

UNIT - IV

Go strings: creating strings, string length, concatenating strings

Go array: declaring array, initializing array, accessing array elements, multi-dimensional array, passing array to function, pointers: how to use pointers, nil pointer, array of pointers, pointer to pointer, passing pointer to function, structure

UNIT - V

Go slice: len() and cap() functions, nil slice, subslicing, append() and copy() functions, Go-range, Go-map, Go-recursion

Go type casting, Go- interfaces, Go-error handling

Books:

1. *An introduction to programming in Go by Caleb Doxsey*
2. *Programming in Go by Mark Summerfield*
3. *The Go programming language by David Chisnall*

Course Code: **CSE412E**	Course Title: **Modeling and Simulation**	Credits: 03 **L - 3 P - 0**

Course Outcomes

- *After successful completion of the course the students would be able to:*
- *Create a relevant model for a multitude of problems from science and engineering by extracting the necessary and relevent information regarding the problem.*
- *Define the different modeling terms by analyzing the system or the data that is present.*
- *Implement the model on the computer and from the results check for the validity of the model and correctness of the assumptions present in the model.*
- *Analyze the outcomes (mostly through visualizations) and make predictions.*

Unit - I

Concepts of Systems, Models, and Simulation. Distributed Lag Model, Cobweb Models, The process of a simulation Study, Exponential Growth Models, Exponential Decay Models.

Unit - II

Type of simulation, Discrete-Event Simulation: Time-Advance Mechanisms, Components and Organization of a Discrete-Event Simulation Model. Monte Carlo Method. Simulation of Single-Server Queuing System, Simulation of an Inventory System

Unit - III

Continuous Simulation: Pure-pursuit Problem.

Random Number Generators: Linear Congruential Generators, Other kinds of Generators, Testing Random-Number Generators. Generating Random Variates: General Approaches, Continuous and Discrete distributions.

Unit - IV

Introduction to GPSS, General Description, GPSS block-diagram, Simulation of a Manufacturing Shop. SNA, Function, Simulation of a Supermarket, GPSS Model of a Simple Telephone System

Unit - V

Output Data Analysis for a Single System: Transient and Steady-State Behaviour of a Stochastic Process, Type of Simulations with regard to output Analysis and Statistical Analysis for Testing Simulation. Verification and Validation of Simulation. An introduction of different types of simulation languages.

Reference books:

1. **G. Gordon,** "*System Simulation*", **Pearson Education**
2. **Law and Kelton,** "*Simulation Modelling and Analysis*", **McGraw Hill**
3. **N. Deo,** "*System Simulation with Digital Computer*", **Prentice Hall of India**
4. **Fred Maryanski,** "*Digital Computer Simulation*", **CBSPD**
5. **James A. Pyne,** "*Introduction to Simulation- Programming Techniques and Methods of Analysis*", **McGraw Hill**
6. **Zeigler and Kim,** "*Theory of Modelling and Simulation*", **Academic Press**
7. **Banks et al,** "*Discrete event Simulation*", Pearson **Education**

B-Tech Computer Science & Engineering
Syllabus for Semester-VIII

Course Code	Course Title	Hours Per Week		Credit
		L	P	
DMS471C	Organisational Behaviour	3	0	3
	Elective –IV	3	0	3
	Elective –V	3	0	3
CSE450C	Project–III	0	20	10
Total Credits		**9**	**20**	**19**

Course Code: **DMS-471C**	Course Title: **Organizational Behaviour**	Credits: 03 **L - 3 P - 2**

Course Outcomes

- *To discuss the scientific foundations of Organisational Behaviour.*
- *Identify personal dimensions of personality, Job satisfaction, motivation and learning.*
- *Analyse group dynamics, teamwork and management process to improve employee performance.*
- *Become familiar with the basics of conflict resolution, stress, power politics and ethics.*
- *Develop understanding about organizational change and development.*

UNIT-I

Organizational Behaviour: Definition, Importance, Historical Background, Fundamental Concepts of OB, Challenges and Opportunities for OB.

Personality and Attitudes: Meaning of personality, Personality Determinants and Traits, Development of Personality, Types of Attitudes, Job Satisfaction.

UNIT-II

Perception: Definition, Nature and Importance, Factors influencing Perception, Perceptual Selectivity, Link between Perception and Decision Making.

Motivation: Definition, Theories of Motivation - Maslow's Hierarchy of Needs Theory, McGregor's Theory X & Y, Herzberg's Motivation-Hygiene Theory, Alderfer's ERG Theory, McClelland's Theory of Needs, Vroom's Expectancy Theory.

UNIT-III

Group Behaviour: Characteristics of Group, Types of Groups, Stages of Group Development, Group Decision Making.

Communication: Communication Process, Direction of Communication, Barriers to Effective Communication.

UNIT-IV

Leadership: Definition, Importance, Theories of Leadership Styles.

Organizational Politics: Definition, Factors contributing to Political Behaviour.

UNIT-V

Conflict Management: Traditional vis-a-vis Modern View of Conflict, Functional and Dysfunctional Conflict, Conflict Process, Negotiation – Bargaining Strategies, Negotiation Process.

Organizational Design: Various Organizational Structures and their Effects on Human Behaviour, Concepts of Organizational Climate and Organizational Culture.

Books Recommended:

1. **Robbins, S. P. & Judge, T.A.: *Organizational Behaviour*, Pearson Education, 15th Edn. 2. Luthans, Fred: Organizational Behaviour, McGraw Hill, 12th Edn.**
2. **Shukla, Madhukar: *Understanding Organizations – Organizational Theory & Practice in India,* PHI**
2. **Fincham, R. & Rhodes, P.: *Principles of Organizational Behaviour*, OUP, 4th Edn.**
3. **Hersey, P., Blanchard, K.H., Johnson, D.E.- *Management of Organizational Behaviour Leading Human Resources*, PHI, 10th Edn.**

Course Code: **CSE450C**	Course Title: **Project III**	Credits: 10 **L - 3 P - 20**

Description

In the 8th semester, students have to take Project III (Major – Project) of 10 credits. Students are advised to extend their Project II to Project III so that they can be strengthen in a particular field which further helps in defining their field of interest / specialization. Students may choose a new topic for Project III without extending Project II by taking prior consent from the allotted supervisor.

An external examiner along with departmental committee evaluate the performance of a student in Project III.

B. Tech. Computer Science and Engineering
Syllabus of Discipline Centric Electives for
Semester Eight

Course Code	Course Title	Hours Per Week		Credits	Pre-requisite	Semester
		L	P			
CSE450E	Big Data	3	0	03	DBMS	8th
CSE451E	Advanced Computer Architecture	3	0	03	CAO	8th
CSE452E	High Performance Computing	3	0	03	Computer Architecture & Organization	8th
CSE453E	Computer Vision	3	0	03	DS	8th
CSE454E	Operation Research	3	0	03	Nill	8th
CSE455E	Deep Learning	3	0	03	Machine Learning/DIP	8th
CSE456E	Ethical Hacking	3	0	03	Computer Network	8th
CSE457E	Cloud Computing	3	0	03	Computer Networks	8th
CSE458E	Natural language processing	3	0	03	Computer Programming	8th
CSE359E	Mobile Application Development	3	0	03	Java	8th
CSE460E	Wireless Communication	3	0	03	Computer Networks/Data Communication	8th
CSE461E	Reinforcement Learning	3	0	03	Machine Learning	8th
CSE462E	Evolutionary Computing	3	0	03	Computer Programming	8th
CSE463E	Software Project Management	3	0	03	Software Engineering	8th
CSE464E	Computational Ethics Design	3	0	03		8th
CSE465E	Bioinformatics	3	0	03	Computer Programming	8th
CSE-466E	Block Chain	3	0	03	Computer Network	8th

Course Code: **CSE450E**	Course Title: **Big Data**	Credits: 03 **L - 3 P - 0**

Course Outcomes

- *Understand the concept of Big Data, challenges faced by it and why classical data analysis tools and techniques are no longer adequate*
- *Get familiar with latest Big Data management tools with main focus on Hadoop ecosystem and Map-reduce*
- *Understand different learning algorithms for analysis of Big Data for knowledge extraction*
- *Model and implement efficient Big Data solutions for various application areas*

UNIT - I

Big Data overview, introduction to the Big Data problem, key enablers for growth of big data, definitions and dimensions of big data, 3Vs and 7Vs of Big Data, current challenges, trends, and applications of Big Data, concept and types of data sets, different sources of big data, different forms of big data

UNIT - II

Mining and learning algorithms that deal with large datasets, Supervised learning: Linear/Logistic regression, Decision trees, Naïve Bayes. Unsupervised learning: K-means clustering, Association rule mining

UNIT - III

Technologies and tools for big data management, Hadoop: Introduction, features, architecture, components and Hadoop ecosystem, Map-Reduce paradigm: proper description and illustration, Basics of stream processing/computing, Data Sciences: introduction, rising and importance

UNIT - IV

Big data analytics in industry verticals, Data analytics lifecycle and methodology: discovery, data preparation, model planning, model building, communicate results, operationalize, Big data value chain

UNIT - V

Data visualization techniques, Text mining: introduction, areas, process and applications, Web mining: introduction, categories and applications, Features and brief description of Amazon Web Services, BlueMix, Cognos, Big insight

Text Books:

1. *Big Data: A Revolution That Will Transform How We Live, Work, and* Think, by Viktor Mayer-Schönberger, Kenneth Cukier.
2. *Hadoop: The Definitive Guide by Tom White (Goodreads Author),* Doug Cutting , O'Reily Publications.
3. *Real-Time Big Data Analytics: Emerging Architecture,* [Kindle Edition], Mike Barlow

Course Code: **CSE451E**	Course Title: ***Advanced Computer Architecture***	Credits: 03 **L - 3 P - 0**

Course Outcomes

- *Provide knowledge on instruction level parallelism*
- *Describe the hardware and software level parallelism to enhance the computing performance*
- *Describe the different operations of performance enhancement such as pipelines, dynamic scheduling, branch prediction, caches, and vector processors*
- *Design the models for arithmetical calculation and to evaluate the CPU performance*
- *Understand the memory hierarchy model for enhanced computing performance*

Unit 1:

Program Execution: compilation, object files, Function call and return, Address space, Data and representation.

Computer Organization: Memory, registers, instruction set architecture, Instruction processing.

Unit 2:

Using processor software approach: Virtual memory, Address translation, paging. Introduction to operating systems, processes, system calls, process management.

Unit 3:

Using processor hardware approach: pipelined processors, structural, data and control hazards, its impact on programming, static and dynamic instruction scheduling, branch handling and prediction.

Unit 4:

Cache memory, organization and its impact on programming, Cache coherence, solutions to cache coherence problem.

Unit 5:

Parallel programming, inter process communication, synchronization, parallel architecture (Fyln's texonomy), shared memory vs message passing, and introduction to programming with message passing with MPI.

Books:

1. *Computer Architecture: A Quantitative Approach*, John L. Hennessy and David A. Patterson, Morgan Kaufmann Publishers Inc.
2. *Computer organization and design: the hardware/software interface*, David A. Patterson and John L. Hennessy, Morgan Kaufmann Publishers Inc.
3. *Advanced Computer Architecture*, Kai Hwang, Tata McGraw- Hill publishing.

Course Code: **CSE452E**	Course Title: ***High Performance Computing***	Credits: 03 **L - 3 P - 0**

Course Outcomes

- *Classify the types of HPC architecture.*
- *Understand the architecture of modern CPU's and how this architecture influences the way programs should be written.*
- *Optimize all aspects in the processes of programming: from compilation, starting and running program by OS, executing (parallel) instructions by CPU, to writing output to disk.*
- *Identify codes suited for parallelizing.*
- *Investigate the performances of parallel code*

Unit - I

Parallel processing: Introduction, Data and Control Parallelism, (concurrency, scalability, speedup, Amolahl's Law), PRAM Model of Parallel Computation, Parallel Algorithm Design.

Unit - II

Multiprocessors and Multicomputer: Processor Organisation. Shared Memory and Message Passing Systems.

Unit - III

Array processing: SIMD Array Processing, Communications, SIMD Interconnection Networks, Algorithms for Array Processing.

Applications: Parallel Algorithms for Matrix Multiplication, Fast Fourier Transform, Linear Systems, Sorting, Numerical Integration, Optimization.

Unit - IV

MPI Commands like: MPI_Init, MPI_Finalize, MPI_Comm_size, MPI_Comm_rank, MPI_Send, MPI_Recv, MPI_Bcast, MPI_Reduce, MPI_Barrier, MPI_Scatter, MPI_Gather

Unit - V

Basic concepts of NVIDIA GPU and CUDA programming CUDA Library, Using CUDA-GDB to debug CUDA program, Using Multiple GPUs, Combining MPI and CUDA

Text Book:

1. Georg Hager, Gerhard Wellein, "*Introduction to High Performance Computing for Scientists and Engineers*", Chapman & Hall / CRC Computational Science series, 2011.

References

- Charles Severance, Kevin Dowd, "*High Performance Computing*", O'Reilly Media, 2nd Edition, 1998.
- Kai Hwang, Faye Alaye Briggs, "*Computer Architecture and Parallel Processing*", McGraw Hill, 1984.
- Pacheco, Peter S., "Parallel Programming with MPI", Morgan Kaufmann Publishers, Inc., California, 1997.
- Cook, Shane. CUDA programming: a developer's guide to parallel computing with GPUs. Newnes, 2012.

Course Code: **CSE453E**	Course Title: ***Computer Vision***	Credits: 03 **L - 3 P - 0**

Course Outcomes

- *Understand the foundations of the modern computer vision theory.*
- *Understand and implement various computer vision algorithms.*
- *Ability to identify, formulate and solve a computer vision problem and evaluate the solution.*
- *Providing a foundation to understand the advanced computer vision theory.*

Unit I: Digital Image Formation and low-level processing

Overview and State-of-the-art, Fundamentals of Image Formation, Transformation: Orthogonal, Euclidean, Affine, Projective, etc.; Fourier Transform, Convolution and Filtering, Image Enhancement, Restoration, Histogram Processing.

Unit II: Feature Extraction

Edge Detection - Derivative of gaussians, Sobel filters Canny edge detector; Features and fitting - RANSAC, Local features, Harris corner detection; High level features – HOG and SIFT

Unit III: Segmentation and Clustering

Region Growing, Edge Based approaches to segmentation, Graph-Cut, Mean-Shift, Agglomerative clustering, K-means, GMMS.

Unit IV: Pattern Classification

Classification pipeline, Visual Bag-of-words. Dimensionality Reduction: PCA, LDA. Neural Nets for classifications. Convolutional Neural Networks for feature detection.

Unit V: Motion Analysis

Background Subtraction and Modeling, Optical Flow, KLT, Spatio-Temporal Analysis, Dynamic Stereo; Motion parameter estimation.

Text Books:

1. *Computer Vision: Algorithms and Applications*" by Richard Szeliski.
2. *Computer Vision- A modern Approach*, by D. Forsyth and J. Ponce, Prentice Hall
3. *Computer Vision- A modern Approach*, by D. Forsyth and J. Ponce, Prentice Hall
4. *Robot Vision*, by B.K.P. Horn, McGraw-Hill.

Reference Books:

- *Introductory Techniques for 3D Computer Vision*, by E Trucco and A. Verri, Prentice Hall

Course Code: **CSE454E**	Course Title: ***Computer Vision***	Credits: 03 **L - 3 P - 0**

Course Outcomes:

- *analyze any real-life system with limited constraints and depict it in a model form.*
- *convert problem into mathematical model.*
- *solve mathematical model manually as well as using software resources.*
- *understand variety of problems such as assignment, transportation, travelling salesman etc.*
- *solve sequencing problems by processing 'n' number of jobs on 'm' number of machines.*
- *find out optimal solution using dynamic programming.*

Unit I: Introduction

Introduction to operation Research, Linear Programming problem. Formulation of LPP, Graphical solution of LPP, simplex method, artificial variables, big-M method.

Unit II: Transportation Problems

Formulation, solution of balanced transportation problem. Finding initial basic feasible solutions. North-west corner rule, least cost method and Vogoles approximation method.

Unit III: Assignment Model and Hungarian method

Assignment Model Formulation, Hungarian method for optimal solution; solving unbalanced problems; travelling salesman problem and assignment.

Unit IV: Sequencing Models

Solution of sequencing problem, processing n jobs through two machines, processing n jobs through three machines, Processing two jobs through m machines.

Unit V: Dynamic Programming

Introduction to Dynamic programming problems, Characteristics and applications of Dynamic Programming, Mathematical formulation and optimal Solution of Dynamic Programming problems.

Books recommended:

1. P. SankaraIyer, *Operations Research*, Tata McGraw Hill 2008
2. A.M. Natarajan, P.Balasubramani, A. Tamilarasi, *Operations*, Pearson Education, 2005.

Course Code: **CSE455E**	Course Title: ***Deep Learning***	Credits: 03 **L - 3 P - 0**

Course Outcomes

- *Identify the deep learning algorithms which are more appropriate for various types of learning tasks in various domains.*
- *understand the theory behind deep learning methods such as Convolutional Neural Networks, Autoencoders and Boltzmann Machines.*
- *Learn deep Turing machines, Apply such deep learning mechanisms to various learning problems. Know the open issues in deep learning, and have a grasp of the current research directions.*
- *Implement deep learning algorithms and solve real-world problems.*

Unit - I:
History, Neural networks basics, McCulloch Pitts Neuron, Thresholding Logic, Perceptrons, Perceptron Learning Algorithm, Multilayer Perceptrons (MLPs), Representation Power of MLPs, Sigmoid Neurons, Gradient Descent, Feedforward Neural Networks, Representation Power of Feedforward Neural Networks

Unit - II:
FeedForward Neural Networks, Backpropagation, Gradient Descent (GD), Momentum Based GD, Nesterov Accelerated GD, Stochastic GD, AdaGrad, RMSProp, Adam, Eigenvalues and eigenvectors, Eigenvalue Decomposition, Basis

Unit - III:
Principal Component Analysis and its interpretations, Singular Value Decomposition, Autoencoders and relation to PCA, Regularization in autoencoders, Regularization: Bias Variance Tradeoff, L2 regularization, Early stopping, Dataset augmentation, Parameter sharing and tying, Injecting noise at input, Ensemble methods, Dropout

Unit IV:
Greedy Layerwise Pre-training, Better activation functions, Better weight initialization methods, Batch Normalization, Convolutional Neural Networks, (Le-Net, Alexnet, Googlenet), Visualizing Convolutional Neural Networks, Guided Backpropagation

Unit V:
Recurrent Neural Networks, Backpropagation through time (BPTT), Vanishing and Exploding Gradients, GRU, LSTMs

Books:

1. *Deep Learning*, MIT Press, Ian Goodfellow and Yoshua Bengio and Aaron Courville http://www.deeplearningbook.org.
2. *Pattern Recognition and Machine Learning*, Bishop, C.M, Springer, 2006.
3. *Artificial Neural Networks*, Yegnanarayana B, PHI Learning Pvt. Ltd, 2009.
4. *Neural Networks and Deep Learning*, Michael Nielsen, online book, http://neuralnetworksanddeeplearning.com
5. *Hands-On Machine Learning* with Scikit-Learn and TensorFlow, O'Reilly, Aurélien Géron.
6. *TensorFlow Deep Learning Cookbook*, Packt, Antonio Gulli and Amita Kapoor.

Course Code: **CSE456E**	Course Title: ***Ethical Hacking***	Credits: 03 **L - 3 P - 0**

Course Outcomes

- *Understand the core concepts related to information security with main focus on hardware and software vulnerabilities, threats, risks, malware, backdoors and key loggers*
- *Learn the underlying principles and techniques associated with ethical hacking and penetration testing*
- *Learn the tools associated with foot printing, concepts of denial of service and social engineering*
- *Understand different approaches as well as related possible attacks of cryptography and steganography and system hacking*
- *Learn the various legal, professional and ethical issues likely to face the domain of ethical hacking and some appropriate tools and techniques associated with ethical hacking*

UNIT I – Introduction to Security & Hacking

Types of Data Stolen from the Organizations, Elements of Information Security, Security Challenges, Vulnerability, Threat, Exploit, Risk, Hacker – Types of Hacker, Role of Security and Penetration Tester, Penetration Testing Methodology, Malicious Software (Malware), Types of Malware, Protection Against Malware, Intruder Attacks on Networks and Computers.

UNIT II - Foot Printing and Social Engineering

Web Tools for Foot Printing, Conducting Competitive Intelligence, Google Hacking, Scanning–Types, Enumeration, Trojans & Backdoors, Virus & Worms, Key Loggers, Denial of Service, Sniffer, Social Engineering – shoulder surfing, Dumpster Diving, Piggybacking.

UNIT III - Data Security

Physical Security – Attacks and Protection, Steganography – Methods, Attacks and Measures, Cryptography – Methods and Types of Attacks, Password Cracking, Wireless Hacking.

UNIT IV- Network Vulnerabilities & Web Application Attacks

Networking & Computer Attacks, Network Sniffing –Types, IP Spoofing, ARP Poisoning, DNS Spoofing, Session Hijacking, Web Server, Web Hacking and Web Application Attacks, SQL Injection, Cross Site Scripting, Cross Site Request Forgery, Buffer Overflow, Email Hacking.

UNIT V - Network Protection System and System Hacking

Routers, Firewall & Honeypots, Intrusion Detection System & Intrusion Prevention System, Proxy & Packet Filtering, Web Filtering, Windows Hacking, Linux Hacking, Introduction to Kali Linux.

Text Books

1. Michael T. Simpson, Kent Backman, James E. "*Corley, Hands -On Ethical Hacking and Network Defense*", Second Edition, CENGAGE Learning, 2010.
2. Rafay Baloch, "*Ethical Hacking and Penetration Testing Guide*", CRC Press, 2015.

References

- Steven DeFino, Barry Kaufman, Nick Valenteen, "*Official Certified Ethical Hacker Review Guide*", CENGAGE Learning, 2009-11-01.
- Patrick Engebretson, "*The Basics of Hacking and Penetration Testing: Ethical Hacking and Penetration Testing Made Easy*", Syngress Basics Series –Elsevier, August 4, 2011.
- Whitaker & Newman, "*Penetration Testing and Network* Defense", Cisco Press, Indianapolis, IN, 2006.
- Kimberly Graves, "*Certified Ethical Hacker Study Guide*", Wiley Publishing, Inc. 2010.
- Kevin Beaver, "*Hacking for Dummies*", Wiley Publishing, Inc. 2013.

Course Code: **CSE457E**	Course Title: ***CLOUD COMPUTING***	Credits: 03 **L - 3 P - 0**

Course Outcomes

- *Articulate the main concepts, key technologies, strength and limitations of cloud computing and possible applications for state-of-the art cloud computing*
- *Identify the architecture and infrastructure of cloud computing*
- *Choose the appropriate technologies, algorithms, and approaches for the related issues.*
- *Understand significance of virtualization in cloud computing*
- *Explain the core issues of cloud computing such as security, privacy, and interoperability*

Unit I - Cloud Introduction

Cloud Computing Fundamentals: Cloud Computing definition, Types of cloud, Cloud services: Benefits and challenges of cloud computing, Evolution of Cloud Computing, usage scenarios and Applications, Business models around Cloud –Major Players in Cloud Computing.

Unit II - Cloud Services

Types of Cloud services: Software as a Service - Platform as a Service –Infrastructure as a Service - Database as a Service - Monitoring as a Service –Communication as services. Service providers- Google App Engine, Amazon EC2, Microsoft Azure, Sales force.

Unit III – Cloud file systems

Introduction to Map Reduce, Information retrieval through Map Reduce, Hadoop File System, GFS, Page Ranking using Map Reduce, Case studies-Ajax, Hadoop.

Unit IV - Virtualization for Cloud

Need for Virtualization – Pros and cons of Virtualization – Types of Virtualization –System VM, Process VM, Virtual Machine monitor – Virtual machine properties -Interpretation and binary translation, HLL VM - Hypervisors – Xen, KVM, VMWare, Virtual Box, Hyper-V.

Unit V - Security, Standards, And Applications

Security in Clouds: Cloud security challenges – Software as a Service Security, Common Standards: The Open Cloud Consortium – The Distributed management Task Force – Standards for application Developers – Standards for Messaging –Standards for Security.

Text Books

1. Bloor R., Kanfman M., Halper F. Judith Hurwitz "*Cloud Computing for Dummies*" (Wiley India Edition), 2010.
2. John Rittinghouse & James Ransome, "*Cloud Computing Implementation Management and Strategy*", CRC Press, 2010.
3. Antohy T Velte, Cloud Computing: "*A Practical Approach*", McGraw Hill,2009
4. James E Smith, Ravi Nair, "*Virtual Machines",* Morgan Kaufmann Publishers,2006.
5. Tom White, *Hadoop: The Definitive Guide*, O'Reilly Media, 2009
6. Jason Venner, *Pro Hadoop*, Apress, 2009

References

- Haley Beard, "*Cloud Computing Best Practices for Managing and Measuring Processes for On-demand Computing*", Applications and Data Centers in the Cloud with SLAs, Emereo Pty Limited, July 2008.
- Michael Miller, Cloud Computing: "*Web-Based Applications That Change the Way You Work and Collaborate Online*", Que Publishing, August 2008.
- Timothy Chou, *Introduction to cloud computing & Business*, Active Book Press, 2010

Course Code: **CSE458E**	Course Title: ***Natural language processing***	Credits: 03 **L - 3 P - 0**

Course Outcomes

- *Practice and learn the computational properties of natural languages and the commonly used algorithms for processing linguistic information*
- *Will learn the techniques widely used in industries*
- *Able to design models and solve the real-world challenges*
- *Learn the concept behind natural language processing and able to extract information from text and verbal data*
- *Understanding semantics and pragmatics of English language for processing*

Unit I: Introduction and Language models

NLP tasks in syntax, semantics and pragmatics, Applications such as information extraction, question answering, and machine translation. The problem of ambiguity. The role of machine learning. Brief history of the field.

The role of language models. Simple N-gram models. Estimating parameters and smoothing. Evaluating language models.

Unit II: Sequence Labelling in NLP

Lexical syntax. Hidden Markov Models (Forward and Viterbi algorithms and EM training). Introduction to vector space models. Bag-of-words. Word embeddings – word2vec and Glove.

Unit III: Deep Learning for NLP

Neural Networks and backpropagation algorithm. CNN for text classification. RNNs – basic BPTT, LSTM and GRU.

Unit IV: Syntactic and Semantic Parsing

Grammar formalisms and treebanks. Efficient parsing for context-free grammars (CFGs). Statistical parsing and probabilistic CFGs (PCFGs). Lexicalized PCFGs. Neural shift-reduce dependency parsing

Lexical semantics and word-sense disambiguation. Compositional semantics. Semantic Role Labeling and Semantic Parsing.

Unit V: Information Extraction

Named entity recognition and relation extraction. IE using sequence labeling. Statistical Machine Translation.

Text Books:

1. Steven Bird, Ewan Klein, Edward Loper, "*Natural Language Processing in Python*" O'Reilly.
2. Jurafsky, D. and J. H. Martin, *Speech and Language Processing*, Prentice-Hall.
3. C. D. and H. Schutze, *Foundations of Statistical Natural Language Processing*, Manning, The MIT Press.
4. Allen,J, *Natural Language Understanding*, The Benajmins Cummings Publishing Company Inc.
5. Cover,T. M. and J.A.Thomas, *Elements of Information Theory*, Wiley.
6. Charniak, E, *Statistical Language Learning*, The MIT Press.

Course Code: **CSE459E**	Course Title: **Mobile Application Development**	Credits: 03 **L -3 P -02**

Course Outcomes:

- *Exposed to technology and business trends impacting mobile applications.*
- *Familiar with mobile operating systems, wireless communications standards and data transmission standards.*
- *Competent with the characterization and architecture of mobile applications.*
- *Competent with understanding enterprise scale requirements of mobile applications.*
- *Competent with designing and developing mobile applications using one application development framework.*

Unit - 1: Introduction

What is Android, Android versions and its feature set. The various Android devices on the market, The Android Market application store, Android Development Environment - System Requirements, Android SDK, Installing Java, and ADT bundle - Eclipse Integrated Development Environment (IDE), Creating Android Virtual Devices (AVDs)

Unit - 2: Android Architecture Overview and Creating an Example Android Application

The Android Software Stack, The Linux Kernel, Android Runtime - Dalvik Virtual Machine, Android Runtime – Core Libraries, Dalvik VM Specific Libraries, Java Interoperability Libraries, Android Libraries, Application Framework, Creating a New Android Project ,Defining the Project Name and SDK Settings, Project Configuration Settings, Configuring the Launcher Icon, Creating an Activity, Running the Application in the AVD, Stopping a Running Application, Modifying the Example Application, Reviewing the Layout and Resource Files.

Unit 3: Android Software Development Platform

Understanding Java SE and the Dalvik Virtual Machine, The Directory Structure of an Android Project, Common Default Resources Folders, The Values Folder, Leveraging Android XML, Screen Sizes, Launching Your Application: The AndroidManifest.xml File, Creating Your First Android Application

Unit 4: Android Framework Overview

Android Application Components, Android Activities: Defining the UI, Android Services: Processing in the Background, Broadcast Receivers: Announcements and Notifications Content Providers: Data Management, Android Intent Objects: Messaging for Components.
Android Manifest XML: Declaring Your Components

Unit 5: Understanding Android Views, View Groups and Layouts

Designing for Different Android Devices, Views and View Groups, Android Layout Managers, The View Hierarchy, Designing an Android User Interface using the Graphical Layout Tool.

Books Recommended:

1. Charlie Collins, Michael Galpin and Matthias Kappler, "*Android in Practice*", DreamTech, 2012
2. AnubhavPradhan, Anil V Despande, *Composing Mobile Apps, Learn, explore, apply*
3. James Dovey and Ash Furrow, "*Beginning Objective C*", Apress, 2012
4. Jeff McWherter and Scott Gowell, "*Professional Mobile Application Development*", Wrox, 2012
5. David Mark, Jack Nutting, Jeff LaMarche and Frederic Olsson, "*Beginning iOS 6 Development: Exploring the iOS SDK*", Apress, 2013.

Course Code: **CSE460E**	Course Title: ***Wireless Communication***	Credits: 03 **L - 3 P - 0**

Course Outcomes

- *Comprehend the knowledge of communication and data communication*
- *Understand the basic cellular concepts including capacity expansion in wireless communication systems, Shadowing and Fading.*
- *Distinguish and compare the performance of different Multiple Access Schemes*
- *Understand the basic principles of GSM, CDMA, WCDMA technologies*
- *Understand the latest trends in wireless communication systems including 4G, 5G, WPAN, LTE and A-LTE.*

UNIT – I

Introduction to Wireless Communication Systems: Evolution of mobile radio communications, paging systems; Cordless telephone systems; overview of generations of cellular systems, Introduction to Personal Communication Services (PCS): PCS architecture.

UNIT -II

Introduction to basic cellular system, Introduction to Wireless Channels and Diversity: Fast Fading Wireless Channel Modeling, Rayleigh/Ricean Fading Channels, BER Performance in Fading Channels, Introduction to Diversity modeling for Wireless Communications.

UNIT - III

2G Networks: Second generation, digital, wireless systems: GSM, IS_136 (D-AMPS), IS-95 CDMA. Global system for Mobile Communication (GSM) system overview: GSM Architecture, Introduction to 2.5G, GPRS, EDGE.

UNIT - IV

Voice signal processing and coding. Spread Spectrum Systems- Cellular code Division Access Systems-Principle, Power Control, effects of multipath propagation on code division multiple access. Mobility Management (Handoff)

UNIT – V

Third Generation (3G) Mobile Services: Introduction to International Mobile Telecommunications 2000 (IMT 2000) vision, Wideband Code Division Multiple Access (W-CDMA), and CDMA 2000, Quality of services in 3G.
Introduction to: UWB, 4G & 5G, Cognitive Radio, Network on a chip.

Text Books:

1. William Stallings "*Wireless Communications & Networks"*
2. Raj Pandya, "*Mobile & Personnel communication Systems and Services*", Prentice Hall India, 2001.
3. Theodore S. Rappaport, "*Wireless Communication- Principles and practices*," 2nd Ed., Pearson Education Pvt. Ltd, 5th Edition, 2008.

Reference Books:

- T.L.Singhal "*Wireless Communication*", Tata McGraw Hill Publication.
- Jochen Schiller, "*Mobile communications*," Pearson Education Pvt. Ltd., 2002.
- Yi –Bing Lin & Imrich Chlamatac, "*Wireless and Mobile Networks Architecture*," John Wiley & Sons,2001.
- Lee, W.C.Y., "*Mobile Cellular Telecommunication*", 2nd Edition, McGraw Hill,1998.
- Smith & Collins, "*3G Wireless Networks*," TMH, 2007
- Schiller, Jochen, "*Mobile Communications*", 2nd Edition, Addison Wesley

Course Code: **CSE461E**	Course Title: **Reinforcement Learning**	Credits: 03 **L - 3 P - 0**

Course Outcomes

- *Understand the elements of an MDP and pose a problem as an MDP.*
- *Understand and appreciate the concepts like policy, value function, reward and the Bellman equations.*
- *Understand the Temporal difference learning and solve an RL problem with TD learning*
- *Appreciate the need and power of Deep RL and solve basic problems using Deep RL.*
- *Get acquainted with the various RL libraries.*

Unit I: Introduction and Language models

Introduction to Reinforcement learning, comparison with supervised and unsupervised learning. The reinforcement learning environment – agent, environment, action and reward. Exploration vs Exploitation – brief idea

Unit II: Markov Decision Processes

Bandit and online problems. Markov Decision Processes. Return and value functions. Dynamic Programming and Bellman equations. Q-Learning

Unit III: Monte Carlo Methods Temporal Difference Learning

Introduction to Monte Carlo estimation. Monte Carlo Policy Evaluation. Gradient Based Monte Carlo Algorithm.

Unit IV: Temporal Difference Learning

Temporal Difference Learning, TD Prediction Methods, Advantages of TD Prediction Methods, Optimality of TD(0), Sarsa: On-policy TD Control, Q-learning: Of-policy TD Control, Expected Sarsa

Unit V: Deep RL and toolkits

Deep Q-Learning. OpenAI gym – modelling and policy learning.

Text Books:

1. *Reinforcement Learning: An Introduction*, Sutton and Barto, 2nd Edition.

Course Code: **CSE462E**	Course Title: ***Evolutionary Computing***	Credits: 03 **L - 3 P - 0**

Course Outcomes

- *Explain the principles underlying Evolutionary Computation in general and Genetic Algorithms in particular and analyze the design of a genetic algorithm, and comment on its weaknesses and strengths.*
- *Demonstrate an understanding of how to apply techniques of swarm intelligence to optimization problems.*
- *Understanding of Memetic Algorithms and its applications and hybridization with GA and a PSO.*
- *Understand how to solve complex search problems with discrete optimization concepts and know various constraint handling methods.*
- *Use of evolutionary algorithms in multi-objective optimization.*

UNIT - I

Genetic Algorithms: GA concepts – encoding, fitness function, population size, selection, crossover and mutation operators, along with the methodologies of applying these operators. Binary GA and their operators, Real Coded GA and their operators.

UNIT - II

Particle Swarm Optimization: PSO Model, global best, Local best, velocity update equations, position update equations, velocity clamping, inertia weight, constriction coefficients, synchronous and asynchronous updates, Binary PSO.

UNIT - III

Memetic Algorithms: Concepts of memes, Incorporating local search as memes, single and multi-memes, hybridization with GA and PSO, Generation Gaps, Performance metrics.

UNIT - IV

Discrete optimization: Use of evolutionary computation to solve travelling salesman problem;Time Table problem,Vehicle routing problem

Constrained optimization: Methods based on rejection strategies,repair strategies,specialized operators and penalty functions.

UNIT - V

Multi-Objective Optimisation: Linear and nonlinear multi-objective problems, convex and non-convex problems, dominance-concepts and properties, Pareto-optimality, Use of Evolutionary computations to solve multi objective optimization, bi level optimization, Theoretical Foundations.

REFERENCES

1. *Recent Advances in Memetic algorithms*, W.E. Hart, N.Krasnogor, J.E Smith, Springer Berlin Heidelberg, New York.
2. *Multi-Objective Optimization using Evolutionary Algorithms*, K. Deb, John Wiley and sons, New Delhi.
3. *Evolutionary Algorithms for solving Multi objective problems*, C.A.Coello Coello, D.A.Van Veldhuizen and G.B Lamont,Kluwer.
4. *Genetic Algorithms and Engineering Design*, M.Gen, and R.Cheng, Wiley,New York
5. *Optimization for engineering Design Algorithms and examples*, K.Deb, Prentice Hall of India, New Delhi
6. *Genetic Algorithms + Data Structures= Evolution Programs*, Z. Michalewicz, Springer-Verlag ,3rd London,UK

Course Code: **CSE463E**	Course Title: **Software Project Management**	Credits: 03 **L - 3 P - 0**

Course Outcomes

- *Define roles and responsibilities by project management process group (initiating, planning, executing controlling, closing).*
- *Articulate the purpose and benefits of project management and defining various metrics of project organization and scheduling.*
- *Demonstrate effective project execution and control techniques that result in successful projects.*
- *Understanding quality assurance metrics and the techniques used for testing the software product. Explain quality management and process improvement in the context of software development projects.*
- *Analyzing software risks and risk management strategies and the tools used for managing the software project.*

Unit-I: Introduction and Software Project Planning

Fundamentals of Software Project Management (SPM), Need Identification, Vision and Scope document, Project Management Cycle, SPM Objectives, Management Spectrum, SPM Framework, Software Project Planning, Planning Objectives, Project Plan, Types of project plan, Structure of a Software Project Management Plan, Software project estimation, Estimation methods, Estimation models, Decision process.

Unit-II: Project Organization and Scheduling

Project Elements, Work Breakdown Structure (WBS), Types of WBS, Functions, Activities and Tasks, Project Life Cycle and Product Life Cycle, Ways to Organize Personnel, Project schedule, Scheduling Objectives, Building the project schedule, Scheduling terminology and techniques, Network Diagrams: PERT, CPM, Bar Charts: Milestone Charts, Gantt Charts.

Unit-III: Project Monitoring and Control

Dimensions of Project Monitoring & Control, Earned Value Analysis, Earned Value Indicators: Budgeted Cost for Work Scheduled (BCWS), Cost Variance (CV), Schedule Variance (SV), Cost Performance Index (CPI), Schedule Performance Index (SPI), Interpretation of Earned Value Indicators, Error Tracking, Software Reviews, Types of Review: Inspections, Deskchecks, Walkthroughs, Code Reviews, Pair Programming.

Unit-IV: Software Quality Assurance and Testing

Testing Objectives, Testing Principles, Test Plans, Test Cases, Types of Testing, Levels of Testing, Test Strategies, Program Correctness, Program Verification & validation, Testing Automation & Testing Tools, Concept of Software Quality, Software Quality Attributes, Software Quality Metrics and Indicators, The SEI Capability Maturity Model CMM), SQA Activities, Formal SQA Approaches: Proof of correctness, Statistical quality assurance, Cleanroom process.

Unit-V: Project Management and Project Management Tools

Software Configuration Management: Software Configuration Items and tasks, Baselines, Plan for Change, Change Control, Change Requests Management, Version Control, Risk Management: Risks and risk types, Risk Breakdown Structure (RBS), Risk Management Process: Risk identification, Risk analysis, Risk planning, Risk monitoring, Cost Benefit Analysis, Software Project Management Tools: CASE Tools, Planning and Scheduling Tools, MS-Project.

References:

1. M. Cotterell, *Software Project Management*, Tata McGraw-Hill Publication.
2. Royce, *Software Project Management*, Pearson Education
3. Kieron Conway, *Software Project Management*, Dreamtech Press
4. S. A. Kelkar, *Software Project Management*, PHI Publication.

Course Code: **CSE464E**	Course Title: **Computational Ethics Design**	Credits: 03 **L - 3 P - 0**

Course Outcomes

- *To acquaint students with a fundamental understanding of various ethical theories.*
- *Explore Artificial Intelligence toward computational design of ethics.*
- *Introduce student to various approaches in machine ethics*
- *Ability to relate ethical concepts to specific engineering problems.*
- *Establish foundation for engaging in further research in the subject*

UNIT - I

Introduction to ethics and its types, metaethics, normative ethics: virtue theories, duty theories, consequentialist theories, introduction to Islamic ethics, applied ethics: normative principles in applied ethics, issues in applied ethics

UNIT - II

Introduction to technoethics, moral agency in technoethics, Value Sensitive Design (VSD), history of VSD, approaches to VSD, VSD critiques, VSD domains, design for values in software development, biased computer system, framework for analyzing bias in computers, minimizing bias in computer system design

UNIT - III

Machine ethics: nature, importance, and difficulty, why machine ethics? philosophical concerns with machine ethics, morality of artificial agents, engineering morality, philosophers, engineers and the design of AMAs, top-down, bottom-up and hybrid approaches, machine manners, ontology agnostic approach

UNIT - IV

Introduction to artificial intelligence (AI), foundations of AI, history of AI, AI applications, intelligent agents and environments, concept of rationality, nature of environments, structure of agents, weak AI, strong AI, ethics of artificial intelligence, ethically aligned design

UNIT - V

Computational models of ethical reasoning: truth-teller, SIROCCO, JEREMY, MedEthEx, LIDA, manners map, ethical ramifications of the Internet of Things

Suggested readings

1. W. Wallach and C. Allen, *Moral machines : teaching robots right from wrong*. Oxford University Press.
2. J. Deigh, *An Introduction to Ethics*. Cambridge University Press.
3. S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach, 3rd edition*. Prentice Hall.

Course Code: **CSE465E**	Course Title: **Bioinformatics**	Credits: 03 **L - 3 P - 0**

Course Outcomes:

- *provide an introduction to what bioinformatics is and why it is important.*
- *extract information from large databases and to use this information in computer modeling.*
- *create an understanding of the intersection of life and information sciences.*
- *speak the language of structure-function relationships, information theory, gene expression.*

UNIT - I

Introduction to bioinformatics and history of bioinformatics. What is bioinformatics? Role of bioinformatics in translational research and clinical practices

UNIT - II

Biological information-categories, acquiring biological information. Biological information analysis-Sequence analysis, Transcription profiling by different methods.

UNIT - III

Biomarker discovery, Next generation sequence and its impact on clinical practice, Biological networks and human diseases.

UNIT - IV

Visual analysis of biological network to understand disease, Genetic variations and personalized medicine.

UNIT - V

Microbiome and human health, Biological information dissemination-Web resources and tools , ethics, security and practice

Textbook/References

1. *Understanding Bioinformatics* by Marketa Zvelebil and Jeremy O. Baum (you may buy it from IU bookstore or **online**)

Course Code: **CSE466E**	Course Title: **Blockchain**	Credits: 03 **L - 3 P - 0**

Course Outcomes:

- *Understand the structure of a blockchain and why/when it is better than a simple distributed database*
- *Evaluate the setting where a blockchain based structure may be applied, its potential and its limitations*
- *Understand what constitutes a "smart" contract, what are its legal implications and what it can and cannot do, now and in the near future*
- *Attain awareness of the new challenges that exist in monetizing businesses around blockchains and smart contracts*

Unit- I

Introduction: Blockchain, Public Ledgers, Blockchain as public ledgers, Bitcoin, Blockchain 2.0, Smart Contracts, Block in a Blockchain, Transactions, Distributed Consensus, The Chain and the Longest Chain, Cryptocurrency to Blockchain 2.0, Permissioned Model of Blockchain.

Basic Crypto Primitives: Cryptographic Hash Function, Properties of a hash function, Hash pointer and Merkle tree, Cryptographic Hash Function, Properties of a hash function, Hash pointer and Merkle tree, Digital Signature, Public Key Cryptography, Basic cryptocurrency

Unit- II

Bitcoin Basics: Creation of coins, Payments and double spending, FORTH (the precursor for Bitcoin scripting), Bitcoin Scripts, Bitcoin P2P Network, Transaction in Bitcoin Network, Block Mining, Block propagation and block relay

Unit- III

Distributed Consensus: Distributed Consensus, Distributed consensus in open environments, Consensus in a Bitcoin network, Proof of Work (PoW), Hashcash PoW, Bitcoin PoW, Attacks on PoW and the monopoly problem, Proof of Stake, Proof of Burn, Proof of Elapsed Time, Mining Difficulty, Mining Pool

Unit- IV

Permissioned Blockchain: Permissioned model and use cases, Design issues for Permissioned blockchains, Consensus models for permissioned blockchain, Distributed consensus in closed environment, Paxos, RAFT Consensus, Byzantine general problem, Byzantine fault tolerant
System, Lamport-Shostak-Pease BFT Algorithm, Practical Byzantine Fault Tolerance, Three phase commit

Unit- V

Blockchain for Enterprise, Blockchain Components and Concepts, Hyperledger, Blockchain use Cases, Blockchain in Financial Service, Blockchain for Trade Logistics, Blockchain in Supply Chain, Blockchain in Government, Blockchain Security, Consensus Scalability, Bitcoin-NG, Byzcoin, Algorand.

Books Recommended:

1. Mastering Bitcoin: Unlocking Digital Cryptocurrencies, by Andreas Antonopoulos, Blockchain by Melanie Swa, O'Reilly.
2. Zero to Blockchain - An IBM Redbooks course, by Bob Dill, David Smits, Hyperledger Fabric.

Syllabus of Open Electives

Course Code	Course Title	Hours Per Week		Credits
		L	P	
CSE001	Introduction to Linux	1	2	02
CSE002	Introduction to Matlab	1	2	02
CSE003	Introduction to Internet and Web Application	1	2	02
CSE004	Basic Computer Applications	1	2	02
CSE005	Computer Fundamentals	1	2	02
CSE006	Latex	1	2	02

Course Code: **CSE 001**	Course Title: **Introduction to Linux**	Credits: 02 **L - 1 P - 2**

Course Outcomes:

- *Understand the basic set of commands and utilities in Linux systems.*
- *learn the important Linux library functions and system calls.*
- *learn and develop software for Linux systems.*
- *understand the inner workings of UNIX-like operating systems.*

UNIT-I

Introduction and Installation

Linux Distributions –Open source software and GNU- Difference between Windows and Linux, Installing Linux in a server configuration, GNOME and KDE– X window system, Managing software.

UNIT-II

Single – Host Administration

Managing users – User text files –User management tools, Command Line, Boot loaders, File Systems, Core System services, Compiling Linux kernel, Linux Firewall.

UNIT- III

Internet Services

Setting up web server using Apache, DNS, FTP, SMTP – Install postfix server, POP and IMAP, Public key Cryptography and SSH, Creating a secure tunnel.

UNIT-IV

Intranet Services

NFS – enable and configure NFS server and client, NIS, SAMBA – Administration, Printing – Install cups – add and manage print jobs, DHCP, Virtualization.

UNIT-V

Linux Process Control & Shell Programming

Linux process environment – login process – parent child relationship – process variable-process monitoring – Invoking foreground and background process – terminating process - Daemons. Introduction to Shell programming – Shell scripts – executing shell scripts - creating scripts – simple examples.

Text Books

1. Wale Soyinka, "*Linux Administration A Beginners Guide"*, 5th edition, Tata McGraw-Hill, 2009.
2. Mc Kinnon, Mc Kinnon, "*Installing and Administrating Linux*", 2nd edition, Wiley, 2004.
3. Steven Graham, Steve Shah, "Linux Administration A Beginners Guide",3rd edition, Dreamtech press , 2003.

References

1. Richard Petersen, "*Linux:The Complete Reference*", 6th edition, Tata McGraw-Hill, 2007.
2. Mark G. Sobell. **"***Practical Guide to Fedora and Red Hat Enterprise Linux"*, 6th Edition, Prentice Hall, 2011.

Course Code: **CSE 002**	Course Title: **MATLAB for Engineers**	Credits: 02 **L - 1 P - 2**

Course Outcomes:

- *Use MATLAB effectively to analyse and visualize data with a clear understanding of the applications of the platform for engineers*
- *Demonstrate understanding and use of fundamental data types, data structures, functions and matrix operations*
- *Apply numeric techniques and computer simulations to solve engineering – related problems*
- *Design and document computer programs and analysis in a careful and complete manner so as to effectively communicate results, to facilitate evaluation and debugging by another programmer, and to anticipate and resolve errors*
- *Create and control simple plots and user – interface graphics objects in MATLAB*

UNIT-I

Introduction to MATLAB and why it is widely used in engineering and science, advantages and limitations of the student edition of MATLAB, Start the MATLAB program and solve simple problems in the command window, Identify and use the various MATLAB windows, Define and use simple matrices, Name and use variables, difference between scalar, array, and matrix calculation[s, Express numbers in either floating-point or scientific notation, Adjust the format used to display numbers in the command window, Save the value of variables used in a MATLAB session, Save a series of commands.

UNIT-II

Built in functions, elementary math functions (common math functions, rounding functions, discrete mathematics functions, trigonometric functions), data analysis functions (maximum and minimum, mean and median, sums and products), sorting functions, random numbers, complex numbers, Recognize and be able to use the special values and functions built into MATLAB.

UNIT-III

Creating Function M-Files, Creating Your Own Toolbox of Functions, Anonymous Functions and Function Handles, Functions, Sub-functions. user defined input, output options, graphical input, Relational and logical operators, Find function, if/else, switch/case structure, for loops, while loops, midpoint break loops.

UNIT-IV

Manipulate matrices, extract data from matrices, solve problems with two matrix variables of different sizes, special matrices, Matrix Operations and Functions, Solutions of Systems of Linear Equations.

UNIT-V

Two-Dimensional Plots, Subplots, Other Types of Two-Dimensional Plots, Three Dimensional Plotting, Editing Plots from the Menu Bar, Creating Plots from the Workspace Window, Saving Your Plots,

List of Experiments

1. Practicing MATLAB environment with simple exercises to familiarize Command Window, History, Workspace, Current Directory, Figure window, Edit window, Shortcuts, Help files.
2. Data types, Constants and Variables, Character constants, operators, Assignment statements.
3. Control Structures: For loops, While, If control structures, Switch, Break, Continue statements.
4. Input-Output functions, Reading and Storing Data.
5. Vectors and Matrices, commands to operate on vectors and matrices, matrix manipulations.
6. Arithmetic operations on Matrices, Relational operations on Matrices, Logical operations on Matrices.

Text Book

1. Holly Moore, "MATLAB for Engineers", Pearson
2. Bansal R.K, Goel A.K., Sharma M.K., "*MATLAB and its Applications in Engineering*", Pearson Education, 2012.

References

1. Amos Gilat, *"MATLAB-An Introduction with Applications*", Wiley India, 2009.
2. Stephen.J.Chapman, *"Programming in MATLAB for Engineers*", Cengage Learning, 2011.

Course Code: **CSE 003**	Course Title: **Introduction to Internet and Web Application**	Credits: 02 **L - 1 P - 2**

Course Outcomes:

- *Apply the basics to design a web page and identify its elements and attributes.*
- *Know all about web search and e-learning and online file storage.*
- *Learn the basics about web browsing by HTTP and URL.*
- *Identify and defend against the risk in system.*

UNIT-I

Internet and its history

Defining and describing internet, brief history;

Internet resources

Email, parts of email, email-address, and email-account; blogs and blogging; chat room; Video conferencing; social networks; wiki-sites; online-games; online shopping/banking.

UNIT-II

WWW

Introduction, Web-browser, Web-server, web-page.

HTML and Web Page Design

Basics of HTML, formatting, fonts, color, hyperlink, image, list, table, Forms.

UNIT-III

Web-services

Introduction, Web-search, mailing, navigation, video-sharing, online file-storage, e-learning.

UNIT-IV

Web-browsing

HTTP, URL, favorites, bookmarks, cookies.

UNIT-V

Protecting the computer

Viruses, worms, Trojan horses, and vulnerabilities, cybercrime: how they work and how they defend against them.

Books Recommended:

1. "*Internet and Web Basics*", by Ned Snell, Bob Temple and T. Michael Clark, "Pearson Education".

2. "*Fundamentals of the Internet and the World Wide Web*", by Raymond Greenlaw and Ellen

Course Code: **CSE 004**	**Course Title:** **Basic Computer Applications**	Credits: 02 **L - 1 P - 2**
Course Outcomes: • *Understand basic organization of computer system.* • *Identify basic peripheral devices and interfaces.* • *Setup and configure computer network.* • *Use basic computer applications.*		

UNIT-I

Computers - An overview of computer and systems
Elements of Computer System, Hardware & Software, Block diagram of a computer, CPU, Memory, Input/Output devices. Mouse and Keyboard, Using a mouse (single and double click and their functions)

UNIT-II

Printers, Scanners, Multi-functional Printer, UPS, Generator,. CD, DVD, USB Drives (Flash / Pen Drives), Identification of different types of cables, Networking devices – Switch, LAN Cable

UNIT-III

Setting Up PC - Connecting each component of computer including LAN, Processes to follow before beginning to work and after completion of work, Concept of Operating System, Types of Operating system.

UNIT-IV

MS Windows XP/ Vista / Windows 7 / Linux - Operating System
a. Start, Shutdown and Restart
b. Desktop, Icons, Recycle Bin, My Computer, My Documents
c. Minimizing, Maximizing, Resizing and Closing Windows
d. Files and folders, directory tree, drives
e. Coping / moving files between folders and drives
f. Renaming, Deleting files and folders
g. Searching, Finding files and folders

UNIT-V

MS Word

a. Creating a new word document
b. Opening an existing document
c. Editing and Saving a document
d. Typing a text, deleting, inserting, finding, replacing, copying and moving text
e. Justifying texts
f. Bold, Italics, Underline, Strike, Double Strike and Colouring text
g. Selecting Font and Font Sizes
h. Formatting page, margins, page size, portrait and landscape
i. Inserting symbols, pictures
j. Using Bullets
k. Using and manipulating tables, inserting / deleting of rows and columns
l. Sorting tables
m. Using Header and footer, Inserting Page number
n. Changing character width and line spacing
o. Printing of a document, Using print preview

Text Books :

1. *Introduction to computers by Peter Norton, Tata McGraw Hill.*

Course Code: **CSE 005**	Course Title: ***COMPUTER FUNDAMENTALS***	Credits: 02 **L - 1 P - 2**

Course Outcomes:

- *Understand introductory concepts of computer*
- *Understand basic organization of computer system*
- *Perform conversion between different number systems*
- *Utilize the Internet Web resources and describe various types of networks*
- *Solve common business problems using appropriate Information Technology applications and systems*

UNIT- I

Introduction to computers, A Simple Computer Model, Hardware and Software essentials of a computer, Need of computer in present world, Characteristics of Computers, Evolution of Computers, Basic Operations of a computer System.

UNIT-II

Input / Output Units: Defining input and output units, types and description of Input –Output devices, Printing devices.

Storage: Primary memory, Memory Cell, Memory organization, ROM, RAM and its types, Secondary storage devices and its types.

UNIT-III

System Software and utilities, Application Software, Licensed and open source software's, Need of Operating Systems, Types of Operating Systems, World Wide Web, How internet works, Benefits and drawbacks of using internet, LAN, WAN, MAN.

UNIT-IV (LAB)

Microsoft office (MS Word, MS Excel, MS Powerpoint), Introduction to Linux operating system.

Text Books :

2. Introduction to computers by Peter Norton, Tata McGraw Hill.

Reference Books:

3. Computer Fundamentals by V. Rajaraman, Pearson Education.
4. Unix concepts and applications, Sumitabha Das, Tata McGraw Hill.

Course Code: **CSE 006**	Course Title: **Latex**	Credits: 02 **L - 1 P - 2**

Course Outcomes:

- *successfully install LaTeX and its related components on a home/personal computer.*
- *use LaTeX and various templates acquired from the course to compose Mathematical documents, presentations, and reports.*
- *access CTAN and other resources to obtain additional LaTeX packages.*
- *Use various methods to either create or import graphics into a LaTeX document.*
- *Use the beamer package to create presentations.*

UNIT- I

What is LATEX? Why Latex? Simple typesetting: Spaces, Quotes, Dashes, Accents, Special symbols, Text positioning; Fonts: Type Style, Type Size

UNIT-II

The Document: Document class, Font and Paper size, Page formats; Page style: Heading declarations, Page numbering, Formatting Lengths; Parts of a Document: Title, Abstract, Chapters, Sections, Subsections, Paragraph etc.; Footnotes and Endnotes

UNIT-III

Making Lists: Bulleted, Numbered, Descriptions and Definitions; Using Tabs: Rows and Columns; Creating tables using the tabular: Enhancements to the tabular, Array package, Multirow package; Using other external packages. The Table environment: Constructing tables, Table Style parameters. Table of Contents, Index and Glossary

UNIT-IV

Advanced Typesetting, Floats and Referencing Typesetting Mathematics: Basics, Superscripts and Subscripts, Mathematical Symbols; Custom commands and operators; Formatting Equations: Numbering and Groups; Typesetting Theorems

UNIT-V

Using Floats: The Figure environment, creating floating figures, Figure placement; using graphics in LATEX: Rotating and Scaling objects. Bibliography: Introduction; Using natbib: basic commands and options, selecting citation style and punctuation; Bibliographic Databases: Using external style files, creating a bibliographic database.