

With effect from 02/08/2016

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD

B.Tech COURSE STRUCTURE (2016-17)
(Common for EEE, ECE, CSE, EIE, BME, IT, ETE, ECM, ICE)

I YEAR I SEMESTER

S. No	Course Code	Course Title	L	T	P	Credits
1	MA101BS	Mathematics-I	3	1	0	3
2	CH102BS	Engineering Chemistry	4	0	0	4
3	PH103BS	Engineering Physics-I	3	0	0	3
4	EN104HS	Professional Communication in English	3	0	0	3
5	ME105ES	Engineering Mechanics	3	0	0	3
6	EE106ES	Basic Electrical and Electronics Engineering	4	0	0	4
7	EN107HS	English Language Communication Skills Lab	0	0	3	2
8	ME108ES	Engineering Workshop	0	0	3	2
9	*EA109MC	NSS	0	0	0	0
		Total Credits	20	1	6	24

I YEAR II SEMESTER

S. No	Course Code	Course Title	L	T	P	Credits
1	PH201BS	Engineering Physics-II	3	0	0	3
2	MA202BS	Mathematics-II	4	1	0	4
3	MA203BS	Mathematics-III	4	1	0	4
4	CS204ES	Computer Programming in C	3	0	0	3
5	ME205ES	Engineering Graphics	2	0	4	4
6	CH206BS	Engineering Chemistry Lab	0	0	3	2
7	PH207BS	Engineering Physics Lab	0	0	3	2
8	CS208ES	Computer Programming in C Lab	0	0	3	2
9	*EA209MC	NCC/NSO	0	0	0	0
		Total Credits	16	2	13	24

* Mandatory Course.

MATHEMATICS- I
(Linear Algebra and Differential Equations)

B.Tech. I Year I Sem.
Course Code: MA101BS

L T/P/D C
3 1/0/0 3

Prerequisites: Foundation course (No prerequisites).

Course Objectives:

To learn

- types of matrices and their properties
- the concept of rank of a matrix and applying the same to understand the consistency
- solving the linear systems
- the concepts of eigen values and eigen vectors and reducing the quadratic forms into their canonical forms
- partial differentiation, concept of total derivative
- finding maxima and minima of functions of two variables
- methods of solving the linear differential equations of first and higher order
- the applications of the differential equations
- formation of the partial differential equations and solving the first order equations.

Course Outcomes:

After learning the contents of this paper the student must be able to

- write the matrix representation of a set of linear equations and to analyze the solution of the system of equations
- find the Eigen values and Eigen vectors which come across under linear transformations
- find the extreme values of functions of two variables with/ without constraints.
- identify whether the given first order DE is exact or not
- solve higher order DE's and apply them for solving some real world problems

UNIT-I

Initial Value Problems and Applications

Exact differential equations - Reducible to exact.

Linear differential equations of higher order with constant coefficients: Non homogeneous terms with RHS term of the type e^{ax} , $\sin ax$, $\cos ax$, polynomials in x , $e^{ax}V(x)$, $xV(x)$ - Operator form of the differential equation, finding particular integral using inverse operator, Wronskian of functions, method of variation of parameters.

Applications: Newton's law of cooling, law of natural growth and decay, Orthogonal trajectories, Electrical circuits.

UNIT-II

Linear Systems of Equations

Types of real matrices and complex matrices, rank, echelon form, normal form, consistency and solution of linear systems (homogeneous and Non-homogeneous) - Gauss elimination, Gauss Jordan and LU decomposition methods- Applications: Finding current in the electrical circuits.

UNIT-III

Eigen values, Eigen Vectors and Quadratic Forms

Eigen values, Eigen vectors and their properties, Cayley - Hamilton theorem (without proof), Inverse and powers of a matrix using Cayley - Hamilton theorem, Diagonalization, Quadratic forms, Reduction of Quadratic forms into their canonical form, rank and nature of the Quadratic forms – Index and signature.

UNIT-IV

Partial Differentiation

Introduction of partial differentiation, homogeneous function, Euler's theorem, total derivative, Chain rule, Taylor's and McLaurin's series expansion of functions of two variables, functional dependence, Jacobian.

Applications: maxima and minima of functions of two variables without constraints and Lagrange's method (with constraints)

UNIT-V

First Order Partial Differential Equations

Formation of partial differential equations by elimination of arbitrary constants and arbitrary functions, Lagrange's method to solve the first order linear equations and the standard type methods to solve the non linear equations.

Text Books:

1. A first course in differential equations with modeling applications by Dennis G. Zill, Cengage Learning publishers.
2. Higher Engineering Mathematics by Dr. B. S. Grewal, Khanna Publishers.

References:

1. Advanced Engineering Mathematics by E. Kreyszig, John Wiley and Sons Publisher.
2. Engineering Mathematics by N. P. Bali, Lakshmi Publications.

ENGINEERING CHEMISTRY

B.Tech. I Year I Sem.
Course Code: CH102BS/CH202BS

L T/P/D C
4 0/0/0 4

Course Objectives:

- 1) To bring adaptability to new developments in Engineering Chemistry and to acquire the skills required to become a perfect engineer.
- 2) To include the importance of water in industrial usage, significance of corrosion control to protect the structures, polymers and their controlled usage.
- 3) To acquire knowledge of engineering materials and about fuels and batteries.
- 4) To acquire required knowledge about engineering materials like cement, refractories and composites.

Course Outcomes:

Students will gain the basic knowledge of electrochemical procedures related to corrosion and its control. They can understand the basic properties of water and its usage in domestic and industrial purposes. They learn the use of fundamental principles to make predictions about the general properties of materials. They can predict potential applications of chemistry and practical utility in order to become good engineers and entrepreneurs.

UNIT-I

Water and its treatment: Introduction – hardness of water – causes of hardness – types of hardness: temporary and permanent – expression and units of hardness – Estimation of hardness of water by complexometric method. Numerical problems. Potable water and its specifications- Steps involved in the treatment of potable water - Disinfection of potable water by chlorination and Ozonization. Defluoridation – Nalgonda technique - Determination of F^- ion by ion- selective electrode method.

Boiler troubles:

sludges, scales and Caustic embrittlement. Internal treatment of Boiler feed water – Calgon conditioning – Phosphate conditioning - Colloidal conditioning – Softening of water by ion-exchange processes. Desalination of water – Reverse osmosis. Numerical problems – Sewage water - Steps involved in treatment of sewage.

UNIT-II

Electrochemistry and Batteries:

Electrochemistry: Electrode- electrode potential, standard electrode potential, types of electrodes – Construction and functioning of Standard hydrogen electrode, calomel and glass electrode. Nernst equation - electrochemical series and its applications. Electrochemical cells: Daniel cell – cell notation, cell reaction and cell emf – Concept of concentration cells – Electrolyte concentration cell – Numerical problems.

Batteries: Cell and battery - Primary battery (dry cell, alkaline cell and Lithium cell) and Secondary battery (lead acid, Ni-Cd and lithium ion cell),

Fuel cells: Hydrogen –oxygen and methanol-oxygen fuel cells – Applications.

UNIT-III

Polymers: Definition – Classification of polymers with examples – Types of polymerization – addition (free radical addition) and condensation polymerization with examples.

Plastics: Definition and characteristics- thermoplastic and thermosetting plastics, compounding and fabrication of plastics (compression and injection moulding). Preparation, Properties and engineering applications of PVC and Bakelite.

Fibers: Characteristics of fibers – preparation, properties and applications of Nylon-6, 6 and Dacron. Fiber reinforced plastics (FRP) – Applications.

Rubbers: Natural rubber and its vulcanization - compounding of rubber.

Elastomers: Characteristics –preparation – properties and applications of Buna-S, Butyl and Thiokol rubber.

Conducting polymers: Characteristics and Classification with examples-mechanism of conduction in trans-polyacetylene and applications of conducting polymers.

Biodegradable polymers: Concept and advantages - Polylactic acid and poly vinyl alcohol and their applications.

UNIT-IV

Fuels and Combustion: Classification- solid fuels: coal – analysis of coal – proximate and ultimate analysis and their significance. Liquid fuels – petroleum and its refining, cracking – types – moving bed catalytic cracking. Knocking – octane and cetane rating, synthetic petrol - Fischer-Tropsch's process; Gaseous fuels – composition and uses of natural gas, LPG and CNG.

Combustion: Definition, Calorific value of fuel – HCV, LCV; Calculation of air quantity required for combustion of a fuel.

UNIT-V

Cement, Refractories, Lubricants and Composites:

Cement: Portland cement, its composition, setting and hardening of Portland cement.

Special cements: White cement, water proof cement, High alumina cement and Acid resistant cement.

Refractories: Classification, characteristics of good refractories, Refractoriness, refractoriness under load, porosity and chemical inertness – applications of refractories.

Lubricants: Classification of lubricants with examples-characteristics of a good lubricants - mechanism of lubrication (thick film, thin film and extreme pressure)- properties of lubricants: viscosity, cloud point, pour point, flash point and fire point.

Composites: Introduction- Constituents of composites – advantages, classification and constituents of composites. Applications of composites.

Text books:

- 1) Engineering Chemistry by P.C. Jain and M. Jain, Dhanpatrai Publishing Company, New Delhi (2010)
- 2) Engineering Chemistry by Rama Devi, Venkata Ramana Reddy and Rath, Cengage learning, New Delhi. (2016)

Reference Books:

- 1) Engineering Chemistry by Shikha Agarwal, Cambridge University Press, Delhi (2015)
- 2) Engineering Chemistry by Shashi Chawla, Dhanpatrai and Company (P) Ltd. Delhi (2011)
- 3) Engineering Chemistry by Thirumala Chary and Laxminarayana, Scitech Publishers, Chennai (2016).

JNTU World

ENGINEERING PHYSICS/ENGINEERING PHYSICS - I

B.Tech. I Year I Sem.
Course Code: PH103BS

L T/P/D C
3 0/0/0 3

Course Objectives:

- To understand interaction of light with matter through interference, diffraction and polarization.
- To able to distinguish ordinary light with a laser light and to realize propagation of light through optical fibers.
- To understand various crystal systems and there structures elaborately.
- To study various crystal imperfections and probing methods like X-RD.

Course outcomes: after completion of this course the student is able to

- Realize the importance of light phenomena in thin films and resolution.
- Learn principle, working of various laser systems and light propagation through optical fibers.
- Distinguish various crystal systems and understand atomic packing factor.
- Know the various defects in crystals.

UNIT-I

Interference: Coherence, division of amplitude and division of wave front, interference in thin films (transmitted and reflected light), Newton's rings experiment.

Diffraction: Distinction between Fresnel and Fraunhofer diffraction, diffraction due to single slit, N-slits, Diffraction grating experiment.

UNIT-II

Polarization: Introduction, Malus's law, double refraction, Nicol prism, Quarter wave and half wave plates.

Lasers: Characteristics of lasers, spontaneous and stimulated emission of radiation, Einstein coefficients, population inversion, ruby laser, helium – neon laser, semi conductor laser, applications of lasers

UNIT-III

Fiber Optics: Principle of optical fiber, construction of fiber, acceptance angle and acceptance cone, numerical aperture, types of optical fibers: step index and graded index fibers, attenuation in optical fibers, applications of optical fibers in medicine and sensors.

UNIT-IV

Crystallography: Space lattice, unit cell and lattice parameters, crystal systems, Bravais lattices, atomic radius, co-ordination number and packing factor of SC, BCC, FCC, HCP and diamond, Miller indices, crystal planes and directions, inter planar spacing of orthogonal crystal systems.

UNIT-V

X-ray Diffraction and Defects in Crystals: Bragg's law, X-ray diffraction methods: Laue method, powder method; point defects: vacancies, substitutional, interstitial, Frenkel and

Schottky defects, line defects (qualitative) and Burger's vector, surface defects: stacking faults, twin, tilt and grain boundaries.

Text Books:

1. Physics Vol. 2, Halliday, Resnick and Kramer John wiley and Sons, Edition 4.
2. Modern Engineering Physics, K. Vijaya Kumar and S. Chandra Lingam, S. Chand and Co. Pvt. Ltd.
3. Introduction to Solid State Physics, Charles Kittel, Wiley Student edition.

Reference Books:

1. X-Ray Crystallography, Phillips, John Wiley publishers.
2. Waves, Frank S Crawford Jr, Berkeley Physics course, Volume 3.
3. Solid State Physics, AJ Dekker, MacMilan Publishers.
4. Introduction to Crystallography, Phillips, John Wiley publishers.

PROFESSIONAL COMMUNICATION IN ENGLISH

B.Tech. I Year I Sem.
Course Code: EN104HS/EN204HS

L T/P/D C
3 0/0/0 3

WILL BE UPLOADED TOMORROW

JNTU World

ENGINEERING MECHANICS

B.Tech. I Year I Sem.
Course Code: ME105ES

L T/P/D C
3 0/0/0 3

Pre Requisites: None

Course Objectives:

- To understand the resolving forces and moments for a given force system
- To analyze the types of friction for moving bodies and problems related to friction.
- To determine the centroid and second moment of area

UNIT-I

Introduction to Mechanics: Basic Concepts, system of Forces Coplanar Concurrent Forces - Components in Space Resultant -Moment of Forces and its Application - Couples and Resultant of Force Systems. Equilibrium of system of Forces: Free body diagrams, Equations of Equilibrium of Coplanar Systems and Spatial Systems.

UNIT-II

Friction: Types of friction -Limiting friction -Laws of Friction -static and Dynamic Frictions - Motion of Bodies –Wedge Screw, Screw-jack and differential screw –jack.

UNIT-III

Centroid and Center of Gravity: Introduction – Centroids of lines – Centroids of area - Centroids of Composite figures - Theorem of Pappus -Centre of Gravity of Bodies – Centroids of Volumes – Center of gravity of composite bodies.

Area moments of Inertia: Introduction – Definition of Moment of Inertia -Polar Moment of Inertia – Radius of gyration. Transfer Theorem for moment of inertia – Moments of inertia by integration - Moments of Inertia of Composite Figures, Product of Inertia, Transfer Formula for Product of Inertia.

UNIT-IV

Mass Moment of Inertia: Introduction - Moment of Inertia of Masses – Radius of gyration - Transfer Formula for Mass Moments of Inertia – Mass moments of inertia by integration - Mass moment of inertia of composite bodies.

Virtual Work: Theory of virtual work-Application.

UNIT-V

Kinetics: Kinetics of a particle-D'Alemberts principle-Motion in a curved path – work, energy and power. Principle of conservation of energy- Kinetics of rigid body in translation, rotation-work done-Principle of work-energy-Impulse-momentum.

Mechanical Vibrations: Definitions, Concepts-Simple Harmonic motion- free vibrations-Simple and compound pendulums

Text Books:

1. Singer's Engineering Mechanics Statics and Dynamics/ K. Vijaya Kumar Reddy, J. Suresh Kumar/ BSP
2. Engineering Mechanics/ Irving Shames, G. Krishna Mohan Rao / Prentice Hall

3. Foundations and applications of Engineering Mechanics by HD Ram and AK Chouhan, Cambridge publications.

References:

1. A Text of Engineering Mechanics /YVD Rao/ K. Govinda Rajulu/ M. Manzoor Hussain / Academic Publishing Company
2. Engineering Mechanics / Bhattacharyya/ Oxford.

JNTU World

BASIC ELECTRICAL AND ELECTRONICS ENGINEERING

B.Tech. I Year I Sem.
Course Code: EE106ES/EE205ES:

L T/P/D C
4 0/0/0 4

Pre-requisite: None

Objectives: Objectives of this course are

- To introduce the concept of electrical circuits and its components
- To introduce the concepts of diodes and transistors, and
- To impart the knowledge of various configurations, characteristics and applications.

OUTCOMES: After this course, the student will be able

- To analyze and solve problems of electrical circuits using network laws and theorems.
- To identify and characterize diodes and various types of transistors.

UNIT- I

Electrical Circuits: R-L-C Parameters, Voltage and Current, Independent and Dependent Sources, Source Transformation – V-I relationship for passive elements, Kirchoff's Laws, Network reduction techniques – series, parallel, series-parallel, star-to-delta, delta-to-star transformation, Nodal Analysis,

Single Phase AC Circuits: R.M.S. and Average values, Form Factor, steady state analysis of series, parallel and series-parallel combinations of R, L and C with sinusoidal excitation, concept of reactance, impedance, susceptance and admittance – phase and phase difference, Concept of power factor, j-notation, complex and polar forms of representation.

UNIT-II

Resonance: Series resonance and Parallel resonance circuits, concept of bandwidth and Q factor, Locus Diagrams for RL, RC and RLC Combinations for Various Parameters.

Network Theorems: Thevenin's, Norton's, Maximum Power Transfer, Superposition, Reciprocity, Tellegen's, Millman's and Compensation theorems for DC and AC excitations.

UNIT- III

P-N Junction Diode: Diode equation, Energy Band diagram, Volt-Ampere characteristics, Temperature dependence, Ideal versus practical, Static and dynamic resistances, Equivalent circuit, Load line analysis, Diffusion and Transition Capacitances.

Rectifiers and Filters: P-N junction as a rectifier - Half Wave Rectifier, Ripple Factor - Full Wave Rectifier, Bridge Rectifier, Harmonic components in Rectifier Circuits, Filters – Inductor Filters, Capacitor Filters, L- section Filters, π - section Filters.

UNIT- IV

Bipolar Junction Transistor (BJT): Construction, Principle of Operation, Symbol, Amplifying Action, Common Emitter, Common Base and Common Collector configurations.

Transistor Biasing And Stabilization - Operating point, DC and AC load lines, Biasing - Fixed Bias, Emitter Feedback Bias, Collector to Emitter feedback bias, Voltage divider bias, Bias

stability, Stabilization against variations in V_{BE} and β , Bias Compensation using Diodes and Transistors.

Transistor Configurations: BJT modeling, Hybrid model, Determination of h-parameters from transistor characteristics, Analysis of CE, CB and CC configurations using h-parameters, Comparison of CE, CB and CC configurations.

UNIT- V

Junction Field Effect Transistor: Construction, Principle of Operation, Symbol, Pinch-Off Voltage, Volt-Ampere Characteristic, Comparison of BJT and FET, Small Signal Model, Biasing FET.

Special Purpose Devices: Breakdown Mechanisms in Semi-Conductor Diodes, Zener diode characteristics, Use of Zener diode as simple regulator, Principle of operation and Characteristics of Tunnel Diode (With help of Energy band diagram) and Varactor Diode, Principle of Operation of SCR.

Text books:

- 1) Basic Electrical and electronics Engineering –M S Sukija TK Nagasarkar Oxford University
- 2) Basic Electrical and electronics Engineering-D P Kothari, I J Nagarath Mc Graw Hill Education

References:

- 1) Electronic Devices and Circuits – R.L. Boylestad and Louis Nashelsky, PEI/PHI, 9th Ed, 2006.
- 2) Millman's Electronic Devices and Circuits – J. Millman and C. C. Halkias, Satyabratajit, TMH, 2/e, 1998.
- 3) Engineering circuit analysis- by William Hayt and Jack E. Kemmerly, Mc Graw Hill Company, 6th edition.
- 4) Linear circuit analysis (time domain phasor and Laplace transform approaches)- 2nd edition by Raymond A. DeCarlo and Pen-Min-Lin, Oxford University Press-2004.
- 5) Network Theory by N. C. Jagan and C. Lakshminarayana, B.S. Publications.
- 6) Network Theory by Sudhakar, Shyam Mohan Palli, TMH.

ENGLISH LANGUAGE COMMUNICATION SKILLS (ELCS) LAB

B.Tech. I Year I Sem.
Course Code: EN107HS/EN207HS

L T/P/D C
0 0/3/0 2

WILL BE UPLOADED TOMORROW

JNTU World

ENGINEERING WORKSHOP

B.Tech. I Year I Sem.
Course Code: ME108ES/ME208ES

L T/P/D C
0 0/3/0 2

Pre-requisites: Practical skill

Course Objective:

- To Study of different hand operated power tools, uses and their demonstration.
- To gain a good basic working knowledge required for the production of various engineering products.
- To provide hands on experience about use of different engineering materials, tools, equipments and processes those are common in the engineering field.
- To develop a right attitude, team working, precision and safety at work place.
- It explains the construction, function, use and application of different working tools, equipment and machines.
- To study commonly used carpentry joints.
- To have practical exposure to various welding and joining processes.
- Identify and use marking out tools, hand tools, measuring equipment and to work to prescribed tolerances.

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

- Study and practice on machine tools and their operations
- Practice on manufacturing of components using workshop trades including plumbing, fitting, carpentry, foundry, house wiring and welding.
- Identify and apply suitable tools for different trades of Engineering processes including drilling, material removing, measuring, chiseling.
- Apply basic electrical engineering knowledge for house wiring practice.

1. TRADES FOR EXERCISES:

At least two exercises from each trade:

- 1) Carpentry
- 2) Fitting
- 3) Tin-Smithy and Development of jobs carried out and soldering.
- 4) Black Smithy
- 5) House-wiring
- 6) Foundry
- 7) Welding
- 8) Power tools in construction, wood working, electrical engineering and mechanical Engineering.

2. TRADES FOR DEMONSTRATION and EXPOSURE:

- Plumbing, Machine Shop, Metal Cutting (Water Plasma)

Text books:

- 1) Workshop Practice /B. L. Juneja / Cengage
- 2) Workshop Manual / K.Venugopal / Anuradha.

Reference books:

- 1) Work shop Manual - P.Kannaiah/ K.L.Narayana/ Scitech
- 2) Workshop Manual / Venkat Reddy/ BSP

JNTU World

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD

B.Tech COURSE STRUCTURE (2016-17)

(Common for EEE, ECE, CSE, EIE, BME, IT, ETE, ECM, ICE)

I YEAR II SEMESTER

S. No	Course Code	Course Title	L	T	P	Credits
1	PH201BS	Engineering Physics-II	3	0	0	3
2	MA202BS	Mathematics-II	4	1	0	4
3	MA203BS	Mathematics-III	4	1	0	4
4	CS204ES	Computer Programming in C	3	0	0	3
5	ME205ES	Engineering Graphics	2	0	4	4
6	CH206BS	Engineering Chemistry Lab	0	0	3	2
7	PH207BS	Engineering Physics Lab	0	0	3	2
8	CS208ES	Computer Programming in C Lab	0	0	3	2
9	*EA209MC	NCC/NSO	0	0	0	0
		Total Credits	16	2	13	24

*Mandatory Course.

PH201BS: ENGINEERING PHYSICS - II

B.Tech. I Year II Sem.

L T/P/D C
3 0/0/0 3

Course Objectives:

- To understand the behavior of a particle quantum mechanically.
- To be able to distinguish pure and impure semi conductors and understand formation of P-N Junction.
- To understand various magnetic and dielectric properties of materials.
- To study super conductor behavior of materials.

Course Outcomes: After completion of this course the student is able to

- Realize the importance of behavior of a particle quantum mechanically.
- Learn concentration estimation of charge carriers in semi conductors.
- Learn various magnetic dielectric properties and apply them in engineering applications.
- Know the basic principles and applications of super conductors.

UNIT - I

Principles of Quantum Mechanics: Waves and particles, de-Broglie hypothesis, matter waves, Davisson and Germer experiment, Heisenberg uncertainty principle, Schrodinger time independent wave equation, physical significance of wave function, particle in 1-D potential box, electron in periodic potential, Kronig-Penny model (qualitative treatment), E-K curve, origin of energy band formation in solids.

UNIT - II

Semiconductor Physics: Fermi level in intrinsic and extrinsic semiconductors, calculation of carrier concentration in intrinsic & extrinsic semiconductors, direct and indirect band gap semiconductors, formation of PN junction, open circuit PN junction, energy diagram of PN junction diode, solar cell: I-V characteristics and applications.

UNIT - III

Dielectric Properties: Electric dipole, dipole moment, dielectric constant, polarizability, electric susceptibility, displacement vector, electronic, ionic and orientation polarizations and calculation of their polarizabilities, internal field, Clausius-Mossotti relation, Piezoelectricity, pyroelectricity and ferroelectricity-BaTiO₃ structure.

UNIT - IV

Magnetic Properties & Superconductivity: Permeability, field intensity, magnetic field induction, magnetization, magnetic susceptibility, origin of magnetic moment, Bohr magneton, classification of dia, para and ferro magnetic materials on the basis of magnetic moment, hysteresis curve based on domain theory, soft and hard magnetic materials, properties of anti-ferro and ferri magnetic materials,

Superconductivity: Superconductivity phenomenon, Meissner effect, applications of superconductivity.

UNIT - V

Introduction to nanoscience: Origin of nanoscience, nanoscale, surface to volume ratio, quantum confinement, dominance of electromagnetic forces, random molecular motion, bottom-up fabrication: Sol-gel, CVD and PVD techniques, top-down fabrication: ball mill method, characterization by XRD, SEM and TEM.

Text Books:

1. Solid State Physics, A. J. Dekkar, Macmillan publishers Ind. Ltd.,
2. Solid State Physics, Chales Kittel, Wiley student edition.
3. Fundamentals of Physics, Alan Giambattisa, BM Richardson and Robert C Richardson, Tata McGraw hill Publishers.

Reference Books:

1. Modern Engineering Physics, K. Vijaya Kumar, S. Chandralingam S. Chand & Co. Pvt. Ltd.,
2. University Physics, Francis W. Sears, Hugh D. Young, Marle Zeemansky and Roger A Freedman, Pearson Education.
3. Fundamentals of Acoustics, Kinster and Frey, John Wiley and Sons.
4. Introduction to Quantum Mechanics Leonard I. Schiff McGraw-Hill

MA102BS/MA202BS: MATHEMATICS - II
(Advanced Calculus)

B.Tech. I Year II Sem.

L T/P/D C
4 1/0/0 4

Prerequisites: Foundation course (No prerequisites).

Course Objectives: To learn

- concepts & properties of Laplace Transforms
- solving differential equations using Laplace transform techniques
- evaluation of integrals using Beta and Gamma Functions
- evaluation of multiple integrals and applying them to compute the volume and areas of regions
- the physical quantities involved in engineering field related to the vector valued functions.
- the basic properties of vector valued functions and their applications to line, surface and volume integrals.

Course Outcomes: After learning the contents of this course the student must be able to

- use Laplace transform techniques for solving DE's
- evaluate integrals using Beta and Gamma functions
- evaluate the multiple integrals and can apply these concepts to find areas, volumes, moment of inertia etc of regions on a plane or in space
- evaluate the line, surface and volume integrals and converting them from one to another

UNIT – I

Laplace Transforms: Laplace transforms of standard functions, Shifting theorems, derivatives and integrals, properties- Unit step function, Dirac's delta function, Periodic function, Inverse Laplace transforms, Convolution theorem (without proof).

Applications: Solving ordinary differential equations (initial value problems) using Laplace transforms.

UNIT - II

Beta and Gamma Functions: Beta and Gamma functions, properties, relation between Beta and Gamma functions, evaluation of integrals using Beta and Gamma functions.

Applications: Evaluation of integrals.

UNIT – III

Multiple Integrals: Double and triple integrals, Change of variables, Change of order of integration. **Applications:** Finding areas, volumes & Center of gravity (evaluation using Beta and Gamma functions).

UNIT – IV

Vector Differentiation: Scalar and vector point functions, Gradient, Divergence, Curl and their physical and geometrical interpretation, Laplacian operator, Vector identities.

UNIT – V

Vector Integration: Line Integral, Work done, Potential function, area, surface and volume integrals, Vector integral theorems: Greens, Stokes and Gauss divergence theorems (without proof) and related problems.

Text Books:

1. Advanced Engineering Mathematics by R K Jain & S R K Iyengar, Narosa Publishers
2. Engineering Mathematics by Srimanthapal and Subodh C. Bhunia, Oxford Publishers

References:

1. Advanced Engineering Mathematics by Peter V. O. Neil, Cengage Learning Publishers.
2. Advanced Engineering Mathematics by Lawrence Turyn, CRC Press

MA203BS: Mathematics - III
(Statistical and Numerical Methods)

B.Tech. I Year II Sem.

L T/P/D C
4 1/0/0 4

Prerequisites: Foundation course (No prerequisites).

Course Objectives: To learn

- random variables that describe randomness or an uncertainty in certain realistic situation
- binomial geometric and normal distributions
- sampling distribution of mean, variance, point estimation and interval estimation
- the testing of hypothesis and ANOVA
- the topics those deals with methods to find roots of an equation
- to fit a desired curve by the method of least squares for the given data
- solving ordinary differential equations using numerical techniques

Course Outcomes: After learning the contents of this course the student must be able to

- differentiate among random variables involved in the probability models which are useful for all branches of engineering
- calculate mean, proportions and variances of sampling distributions and to make important decisions s for few samples which are taken from a large data
- solve the tests of ANOVA for classified data
- find the root of a given equation and solution of a system of equations
- fit a curve for a given data
- find the numerical solutions for a given first order initial value problem

UNIT – I

Random variables and Distributions:

Introduction, Random variables, Discrete random variable, Continuous random variable, Probability distribution function, Probability density function, Expectation, Moment generating function, Moments and properties. Discrete distributions: Binomial and geometric distributions. Continuous distribution: Normal distributions.

UNIT – II

Sampling Theory: Introduction, Population and samples, Sampling distribution of means (σ Known)-Central limit theorem, t-distribution, Sampling distribution of means (σ unknown)-Sampling distribution of variances – χ^2 and F- distributions, Point estimation, Maximum error of estimate, Interval estimation.

UNIT – III

Tests of Hypothesis: Introduction, Hypothesis, Null and Alternative Hypothesis, Type I and Type II errors, Level of significance, One tail and two-tail tests, Tests concerning one mean and proportion, two means-proportions and their differences-ANOVA for one-way classified data.

UNIT – IV

Algebraic and Transcendental Equations & Curve Fitting: Introduction, Bisection Method, Method of False position, Iteration methods: fixed point iteration and Newton Raphson methods. Solving linear system of equations by Gauss-Jacobi and Gauss-Seidal Methods.

Curve Fitting: Fitting a linear, second degree, exponential, power curve by method of least squares.

UNIT – V

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

Text Books:

1. Probability and Statistics for Engineers by Richard Arnold Johnson, Irwin Miller and John E. Freund, New Delhi, Prentice Hall.
2. Probability and Statistics for Engineers and Sciences by Jay L. Devore, Cengage Learning.
3. Numerical Methods for Scientific and Engineering Computation by M. K. Jain, S. R. K. Iyengar and R. K. Jain, New Age International Publishers

References:

1. Fundamentals of Mathematical Statistics by S. C. Gupta & V. K. Kapoor, S. Chand.
2. Introductory Methods of Numerical Analysis by S. S. Sastry, PHI Learning Pvt. Ltd.
3. Mathematics for engineers and scientists by Alan Jeffrey, 6th edition, CRC press.

CS104ES/CS204ES: COMPUTER PROGRAMMING IN C

B.Tech. I Year II Sem.

L T/P/D C
3 0/0/0 3

Course Objectives:

- To learn the fundamentals of computers.
- To understand the various steps in Program development.
- To learn the syntax and semantics of C Programming Language.
- To learn how to write modular and readable C Programs.
- To learn to write programs using structured programming approach in C to solve problems.

Course Outcomes:

- Demonstrate the basic knowledge of computer hardware and software.
- Ability to write algorithms for solving problems.
- Ability to draw flowcharts for solving problems.
- Ability to code a given logic in C programming language.
- Gain knowledge in using C language for solving problems.

UNIT - I

Introduction to Computers – Computer Systems, Computing Environments, Computer Languages, Creating and running programs, Program Development, algorithms and flowcharts , Number systems-Binary, Decimal, Hexadecimal and Conversions, storing integers and real numbers.

Introduction to C Language – Background, C Programs, Identifiers, Types, Variables, Constants, Input / Output, Operators(Arithmetic, relational, logical, bitwise etc.), Expressions, Precedence and Associativity, Expression Evaluation, Type conversions, Statements- Selection Statements(making decisions) – if and switch statements, Repetition statements (loops)-while, for, do-while statements, Loop examples, other statements related to looping – break, continue, goto, Simple C Program examples.

UNIT - II

Functions-Designing Structured Programs, Functions, user defined functions, inter function communication, Standard functions, Scope, Storage classes-auto, register, static, extern, scope rules, type qualifiers, recursion- recursive functions, Limitations of recursion, example C programs.

Arrays – Concepts, using arrays in C, inter function communication, array applications- linear search, binary search and bubble sort, two – dimensional arrays, multidimensional arrays, C program examples.

UNIT - III

Pointers – Introduction (Basic Concepts), Pointers for inter function communication, pointers to pointers, compatibility, Pointer Applications-Arrays and Pointers, Pointer Arithmetic and

arrays, Passing an array to a function, memory allocation functions, array of pointers, programming applications, pointers to void, pointers to functions.

Strings – Concepts, C Strings, String Input / Output functions, arrays of strings, string manipulation functions, string / data conversion, C program examples.

UNIT - IV

Enumerated, Structure and Union Types – The Type Definition (typedef), Enumerated types, Structures –Declaration, initialization, accessing structures, operations on structures, Complex structures-Nested structures, structures containing arrays, structures containing pointers, arrays of structures, structures and functions, Passing structures through pointers, self referential structures, unions, bit fields, C programming examples, command–line arguments, Preprocessor commands.

UNIT – V

Input and Output – Concept of a file, streams, text files and binary files, Differences between text and binary files, State of a file, Opening and Closing files, file input / output functions (standard library input / output functions for files), file status functions (error handling), Positioning functions (fseek ,rewind and ftell), C program examples.

Text Books:

1. Computer Science: A Structured Programming Approach Using C, B. A. Forouzan and R. F. Gilberg, Third Edition, Cengage Learning.
2. Programming in C. P. Dey and M Ghosh , Second Edition, Oxford University Press.

Reference Books:

1. The C Programming Language, B.W. Kernighan and Dennis M. Ritchie, Second Edition, Pearson education.
2. Programming with C, B. Gottfried, 3rd edition, Schaum's outlines, McGraw Hill Education (India) Pvt Ltd.
3. C From Theory to Practice, G S. Tselikis and N D. Tselikas, CRC Press.
4. Basic computation and Programming with C, Subrata Saha and S. Mukherjee, Cambridge University Press.

ME106ES/ME205ES: ENGINEERING GRAPHICS

B.Tech. I Year II Sem.

L T/P/D C

2 0/0/4 4

Pre-requisites: None

Course objectives:

- To provide basic concepts in engineering drawing.
- To impart knowledge about standard principles of orthographic projection of objects.
- To draw sectional views and pictorial views of solids.

Course Outcomes:

- Ability to prepare working drawings to communicate the ideas and information.
- Ability to read, understand and interpret engineering drawings.

UNIT – I

Introduction To Engineering Drawing: Principles of Engineering Graphics and their Significance, Conic Sections including the Rectangular Hyperbola – General method only. Cycloid, Epicycloid and Hypocycloid Involute. Scales – Plain, Diagonal, and Vernier Scales.

UNIT - II

Orthographic Projections: Principles of Orthographic Projections – Conventions – Projections of Points and Lines Projections of Plane regular geometric figures.—Auxiliary Planes.

UNIT – III

Projections of Regular Solids – Auxiliary Views.

UNIT – IV

Sections or Sectional views of Right Regular Solids – Prism, Cylinder, Pyramid, Cone – Auxiliary views – Sections of Sphere. Development of Surfaces of Right Regular Solids – Prism, Cylinder, Pyramid, and Cone

UNIT – V

Isometric Projections: Principles of Isometric Projection – Isometric Scale – Isometric Views – Conventions – Isometric Views of Lines, Plane Figures, Simple and Compound Solids – Isometric Projection of objects having non- isometric lines. Isometric Projection of Spherical Parts. Conversion of Isometric Views to Orthographic Views and Vice-versa – Conventions Auto CAD: Basic principles only.

Text Books:

1. Engineering Drawing / Basant Agrawal and Mc Agrawal/ Mc Graw Hill
2. Engineering Drawing/ M.B. Shah, B.C. Rane / Pearson.

Reference Books:

1. Engineering Drawing / N.S. Parthasarathy and Vela Murali/ Oxford
2. Engineering Drawing N.D. Bhatt / Charotar

CH206BS: ENGINEERING CHEMISTRY LAB

B.Tech. I Year II Sem.

L T/P/D C
0 0/3/0 2

LIST OF EXPERIMENTS

Volumetric Analysis:

1. Estimation of Ferrous ion by Dichrometry.
2. Estimation of hardness of water by Complexometric method using EDTA.
3. Estimation of Ferrous and Ferric ions in a given mixture by Dichrometry.
4. Estimation Ferrous ion by Permanganometry.
5. Estimation of copper by Iodomery.
6. Estimation of percentage of purity of MnO_2 in pyrolusite
7. Determination of percentage of available chlorine in bleaching powder.
8. Determination of salt concentration by ion- exchange resin.

Instrumental methods of Analysis:

1. Estimation of HCl by Conductometry.
2. Estimation of Ferrous ion by Potentiometry.
3. Determination of Ferrous iron in cement by Colorimetric method.
4. Determination of viscosity of an oil by Redwood / Oswald's Viscometer.
5. Estimation of manganese in $KMnO_4$ by Colorimetric method.
6. Estimation of HCl and Acetic acid in a given mixture by Conductometry.
7. Estimation of HCl by Potentiometry.

Preparation of Polymers:

1. Preparation of Bakelite and urea formaldehyde resin.

Note: All the above experiments must be performed.

Text Books:

1. Vogel's Text Book of Quantitative Chemical Analysis, 5th Edition (2015)
2. G. H. Jeffery, J. Bassett, J. Mendham and R. C. Denney.
3. A Text Book on experiments and calculations in Engineering Chemistry by S.S. Dara S. Chand & Company Ltd., Delhi (2003).

PH107BS/PH207BS: ENGINEERING PHYSICS LAB

B.Tech. I Year II Sem.

L T/P/D C
0 0/3/0 2

LIST OF EXPERIMENTS

1. Dispersive power of the material of a prism – Spectrometer.
2. Determination of wavelengths of white source – Diffraction grating.
3. Newton's Rings – Radius of curvature of Plano convex lens.
4. Melde's experiment – Transverse and longitudinal modes.
5. Charging, discharging and time constant of an R-C circuit.
6. L-C-R circuit – Resonance & Q-factor.
7. Magnetic field along the axis of current carrying coil – Stewart and Gees method and to verify Biot – Savart's law.
8. Study the characteristics of LED and LASER diode.
9. Bending losses of fibres & Evaluation of numerical aperture of a given fibre.
10. Energy gap of a material of p-n junction.
11. Torsional pendulum – Rigidity modulus.
12. Wavelength of light, resolving power and dispersive power of a diffraction grating using laser.
13. V-I characteristics of a solar cell.

Note: Minimum 10 experiments must be performed.

CS108ES/CS208ES: COMPUTER PROGRAMMING IN C LAB

B.Tech. I Year II Sem.

L T/P/D C
0 0/3/0 2

Course Objective:

- To write programs in C using structured programming approach to solve the problems.

Course Outcomes:

- Ability to design and test programs to solve mathematical and scientific problems.
- Ability to write structured programs using control structures and functions.

Recommended Systems/Software Requirements:

- Intel based desktop PC
- GNU C Compiler

- a) Write a C program to find the factorial of a positive integer.
 - b) Write a C program to find the roots of a quadratic equation.
- a) Write a C program to determine if the given number is a prime number or not.
 - b) A Fibonacci sequence is defined as follows: the first and second terms in the sequence are 0 and 1. Subsequent terms are found by adding the preceding two terms in the sequence. Write a C program to generate the first n terms of the sequence.
- a) Write a C program to construct a pyramid of numbers.
 - b) Write a C program to calculate the following Sum:
$$\text{Sum} = 1 - x^2/2! + x^4/4! - x^6/6! + x^8/8! - x^{10}/10!$$
- a) The least common multiple (LCM) of two positive integers a and b is the smallest integer that is evenly divisible by both a and b. Write a C program that reads two integers and calls LCM (a, b) function that takes two integer arguments and returns their LCM. The LCM (a, b) function should calculate the least common multiple by calling the GCD (a, b) function and using the following relation:
$$\text{LCM}(a, b) = ab / \text{GCD}(a, b)$$
 - b) Write a C program that reads two integers n and r to compute the ncr value using the following relation:
$$n_{c_r}(n, r) = n! / r! (n-r)! .$$
 Use a function for computing the factorial value of an integer.
- a) Write C program that reads two integers x and n and calls a recursive function to compute x^n
 - b) Write a C program that uses a recursive function to solve the Towers of Hanoi problem.
 - c) Write a C program that reads two integers and calls a recursive function to compute n_{c_r} value.

6. a) Write a C program to generate all the prime numbers between 1 and n, where n is a value supplied by the user using Sieve of Eratosthenes algorithm.
b) Write a C program that uses non recursive function to search for a Key value in a given list of integers. Use linear search method.
7. a) Write a menu-driven C program that allows a user to enter n numbers and then choose between finding the smallest, largest, sum, or average. The menu and all the choices are to be functions. Use a switch statement to determine what action to take. Display an error message if an invalid choice is entered.
b) Write a C program that uses non recursive function to search for a Key value in a given sorted list of integers. Use binary search method.
8. a) Write a C program that implements the Bubble sort method to sort a given list of integers in ascending order.
b) Write a C program that reads two matrices and uses functions to perform the following:
 1. Addition of two matrices
 2. Multiplication of two matrices
9. a) Write a C program that uses functions to perform the following operations:
 1. to insert a sub-string into a given main string from a given position.
 2. to delete n characters from a given position in a given string.
b) Write a C program that uses a non recursive function to determine if the given string is a palindrome or not.
10. a) Write a C program to replace a substring with another in a given line of text.
b) Write a C program that reads 15 names each of up to 30 characters, stores them in an array, and uses an array of pointers to display them in ascending (ie. alphabetical) order.
11. a) 2's complement of a number is obtained by scanning it from right to left and complementing all the bits after the first appearance of a 1. Thus 2's complement of 11100 is 00100. Write a C program to find the 2's complement of a binary number.
b) Write a C program to convert a positive integer to a roman numeral. Ex. 11 is converted to XI.
12. a) Write a C program to display the contents of a file to standard output device.
b) Write a C program which copies one file to another, replacing all lowercase characters with their uppercase equivalents.
13. a) Write a C program to count the number of times a character occurs in a text file. The file name and the character are supplied as command-line arguments.
b) Write a C program to compare two files, printing the first line where they differ.
14. a) Write a C program to change the nth character (byte) in a text file. Use fseek function.

- b) Write a C program to reverse the first n characters in a file. The file name and n are specified on the command line. Use fseek function.
15. a) Write a C program to merge two files into a third file (i.e., the contents of the first file followed by those of the second are put in the third file).
- b) Define a macro that finds the maximum of two numbers. Write a C program that uses the macro and prints the maximum of two numbers.

Reference Books:

1. Mastering C, K.R. Venugopal and S.R. Prasad, TMH Publishers.
2. Computer Programming in C, V. Rajaraman, PHI.
3. Programming in C, Stephen G. Kochan, Fourth Edition, Pearson Education.
4. C++: The complete reference, H. Schildt, TMH Publishers.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD

B.TECH. COMPUTER SCIENCE AND ENGINEERING

COURSE STRUCTURE & SYLLABUS (2016-17)

II YEAR I SEMESTER

S. No	Course Code	Course Title	L	T	P	Credits
1	MA301BS	Mathematics – IV	4	1	0	4
2	CS302ES	Data Structures through C++	4	0	0	4
3	CS303ES	Mathematical Foundations of Computer Science	4	0	0	4
4	CS304ES	Digital Logic Design	3	0	0	3
5	CS305ES	Object Oriented Programming through Java	3	0	0	3
6	CS306ES	Data Structures through C++ Lab	0	0	3	2
7	CS307ES	IT Workshop	0	0	3	2
8	CS308ES	Object Oriented Programming through Java Lab	0	0	3	2
9	* MC300ES	Environmental Science and Technology	3	0	0	0
		Total Credits	21	1	9	24

II YEAR II SEMESTER

S. No	Course Code	Course Title	L	T	P	Credits
1	CS401BS	Computer Organization	4	0	0	4
2	CS402ES	Database Management Systems	4	0	0	4
3	CS403ES	Operating Systems	4	0	0	4
4	CS404ES	Formal Languages and Automata Theory	3	0	0	3
5	SM405MS	Business Economics and Financial Analysis	3	0	0	3
6	CS406ES	Computer Organization Lab	0	0	3	2
7	CS407ES	Database Management Systems Lab	0	0	3	2
8	CS408ES	Operating Systems Lab	0	0	3	2
9	* MC400HS	Gender Sensitization Lab	0	0	3	0
		Total Credits	18	0	12	24

* Satisfactory/Unsatisfactory

MA301BS: MATHEMATICS - IV
(Complex Variables and Fourier Analysis)

B.Tech. II Year I Sem.

L	T	P	C
4	1	0	4

Prerequisites: Foundation course (No Prerequisites).

Course Objectives: To learn

- differentiation and integration of complex valued functions
- evaluation of integrals using Cauchy's integral formula
- Laurent's series expansion of complex functions
- evaluation of integrals using Residue theorem
- express a periodic function by Fourier series and a non-periodic function by Fourier transform
- to analyze the displacements of one dimensional wave and distribution of one dimensional heat equation

Course Outcomes: After learning the contents of this paper the student must be able to:

- analyze the complex functions with reference to their analyticity, integration using Cauchy's integral theorem
- find the Taylor's and Laurent's series expansion of complex functions
- the bilinear transformation
- express any periodic function in term of sines and cosines
- express a non-periodic function as integral representation
- analyze one dimensional wave and heat equation

UNIT – I

Functions of a complex variable: Introduction, Continuity, Differentiability, Analyticity, properties, Cauchy, Riemann equations in Cartesian and polar coordinates. Harmonic and conjugate harmonic functions-Milne-Thompson method

UNIT - II

Complex integration: Line integral, Cauchy's integral theorem, Cauchy's integral formula, and Generalized Cauchy's integral formula, Power series: Taylor's series- Laurent series, Singular points, isolated singular points, pole of order m – essential singularity, Residue, Cauchy Residue theorem (Without proof).

UNIT – III

Evaluation of Integrals: Types of real integrals:

a) Improper real integrals $\int_{-\infty}^{\infty} f(x)dx$ (b) $\int_c^{c+2\pi} f(\cos \theta, \sin \theta)d\theta$

Bilinear transformation- fixed point- cross ratio- properties- invariance of circles.

UNIT – IV

Fourier series and Transforms: Introduction, Periodic functions, Fourier series of periodic function, Dirichlet's conditions, Even and odd functions, Change of interval, Half range sine and cosine series.

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

UNIT – V

Applications of PDE: Classification of second order partial differential equations, method of separation of variables, Solution of one dimensional wave and heat equations.

TEXT BOOKS:

1. A first course in complex analysis with applications by Dennis G. Zill and Patrick Shanahan, Johns and Bartlett Publishers.
2. Higher Engineering Mathematics by Dr. B. S. Grewal, Khanna Publishers.
3. Advanced engineering Mathematics with MATLAB by Dean G. Duffy

REFERENCES:

1. Fundamentals of Complex Analysis by Saff, E. B. and A. D. Snider, Pearson.
2. Advanced Engineering Mathematics by Louis C. Barrett, McGraw Hill.

CS302ES: DATA STRUCTURES THROUGH C++

B.Tech. II Year I Sem.

L T P C
4 0 0 4

Course Objectives:

- To understand the basic concepts such as Abstract Data Types, Linear and Non Linear Data structures.
- To understand the notations used to analyze the Performance of algorithms.
- To understand the behavior of data structures such as stacks, queues, trees, hash tables, search trees, Graphs and their representations.
- To choose an appropriate data structure for a specified application.
- To understand and analyze various searching and sorting algorithms.
- To learn to implement ADTs such as lists, stacks, queues, trees, graphs, search trees in C++ to solve problems.

Course Outcomes:

- Ability to choose appropriate data structures to represent data items in real world problems.
- Ability to analyze the time and space complexities of algorithms.
- Ability to design programs using a variety of data structures such as stacks, queues, hash tables, binary trees, search trees, heaps, graphs, and B-trees.
- Able to analyze and implement various kinds of searching and sorting techniques.

UNIT - I

C++ Programming Concepts: Review of C, input and output in C++, functions in C++-value parameters, reference parameters, Parameter passing, function overloading, function templates, Exceptions-throwing an exception and handling an exception, arrays, pointers, new and delete operators, class and object, access specifiers, friend functions, constructors and destructor, Operator overloading, class templates, Inheritance and Polymorphism..

Basic Concepts - Data objects and Structures, Algorithm Specification-Introduction, Recursive algorithms, Data Abstraction, Performance analysis- time complexity and space complexity, Asymptotic Notation-Big O, Omega and Theta notations, Complexity Analysis Examples, Introduction to Linear and Non Linear data structures.

UNIT - II

Representation of single, two dimensional arrays, sparse matrices-array and linked representations.

Linear list ADT-array representation and linked representation, Singly Linked Lists-Operations-Insertion, Deletion, Circularly linked lists-Operations for Circularly linked lists, Doubly Linked Lists- Operations- Insertion, Deletion.

Stack ADT, definition, array and linked implementations, applications-infix to postfix conversion, Postfix expression evaluation, recursion implementation, Queue ADT, definition, array and linked Implementations, Circular queues-Insertion and deletion operations.

UNIT - III

Trees – definition, terminology, Binary trees-definition, Properties of Binary Trees, Binary Tree ADT, representation of Binary Trees-array and linked representations, Binary Tree traversals, Threaded binary trees, Priority Queues –Definition and applications, Max Priority Queue ADT-implementation-Max Heap-Definition, Insertion into a Max Heap, Deletion from a Max Heap.

UNIT - IV

Searching - Linear Search, Binary Search, Hashing-Introduction, hash tables, hash functions, Overflow Handling, Comparison of Searching methods.

Sorting-Insertion Sort, Selection Sort, Radix Sort, Quick sort, Heap Sort, Merge sort, Comparison of Sorting methods.

.

UNIT - V

Graphs–Definitions, Terminology, Applications and more definitions, Properties, Graph ADT, Graph Representations- Adjacency matrix, Adjacency lists, Graph Search methods - DFS and BFS, Complexity analysis,

Search Trees-Binary Search Tree ADT, Definition, Operations- Searching, Insertion and Deletion, Balanced search trees-AVL Trees-Definition and Examples only, B-Trees-Definition and Examples only, Red-Black Trees-Definitions and Examples only, Comparison of Search Trees.

TEXT BOOKS:

1. Data structures, Algorithms and Applications in C++, 2nd Edition, Sartaj Sahni, Universities Press.
2. Data structures and Algorithms in C++, Adam Drozdek, 4th edition, Cengage learning.

REFERENCE BOOKS:

1. Data structures with C++, J. Hubbard, Schaum's outlines, TMH.
2. Data structures and Algorithms in C++, M.T. Goodrich, R. Tamassia and D. Mount, Wiley India.
3. Data structures and Algorithm Analysis in C++, 3rd edition, M. A. Weiss, Pearson.
4. Classic Data Structures, D. Samanta, 2nd edition, PHI.

CS303ES: MATHEMATICAL FOUNDATIONS OF COMPUTER SCIENCE

B.Tech. II Year I Sem.

L	T	P	C
4	0	0	4

Course Objectives:

- To introduce the concepts of mathematical logic.
- To introduce the concepts of sets, relations, and functions.
- To perform the operations associated with sets, functions, and relations.
- To relate practical examples to the appropriate set, function, or relation model, and interpret the associated operations and terminology in context.
- To introduce generating functions and recurrence relations.
- To use Graph Theory for solving problems.

Course Outcomes

- Ability to apply mathematical logic to solve problems.
- Understand sets, relations, functions, and discrete structures.
- Able to use logical notation to define and reason about fundamental mathematical concepts such as sets, relations, and functions.
- Able to formulate problems and solve recurrence relations.
- Able to model and solve real-world problems using graphs and trees.

UNIT - I

Mathematical logic: Introduction, Statements and Notation, Connectives, Normal Forms, Theory of Inference for the Statement Calculus, The Predicate Calculus, Inference Theory of the Predicate Calculus.

UNIT - II

Set theory: Introduction, Basic Concepts of Set Theory, Representation of Discrete Structures, Relations and Ordering, Functions.

Algebraic Structures: Introduction, Algebraic Systems, Semi groups and Monoids, Groups, Lattices as Partially Ordered Sets, Boolean algebra.

UNIT - III

Elementary Combinatorics: Basics of Counting, Combinations and Permutations, Enumeration of Combinations and Permutations, Enumerating Combinations and Permutations with Repetitions, Enumerating Permutations with Constrained Repetitions, Binomial Coefficients, The Binomial and Multinomial Theorems, The Principle of Inclusion-Exclusion.

UNIT - IV

Recurrence Relations: Generating Functions of Sequences, Calculating Coefficients of generating functions, Recurrence relations, Solving recurrence relations by substitution and

Generating functions, The method of Characteristic roots, Solutions of Inhomogeneous Recurrence Relations.

UNIT - V

Graphs: Basic Concepts, Isomorphisms and Subgraphs, Trees and their Properties, Spanning Trees, Directed Trees, Binary Trees, Planar Graphs, Euler's Formula, Multigraphs and Euler Circuits, Hamiltonian Graphs, Chromatic Numbers, The Four-Color Problem.

TEXT BOOKS:

1. Discrete Mathematical Structures with Applications to Computer Science, J.P. Tremblay, R. Manohar, McGraw Hill education (India) Private Limited. (**UNITS - I , II**)
2. Discrete Mathematics for Computer Scientists & Mathematicians, Joe L. Mott, Abraham Kandel, Theodore P. Baker, Pearson , 2nd ed. (**Units - III, IV, V**)

REFERENCE BOOKS:

1. Discrete Mathematics and its Applications, Kenneth H. Rosen, 7th Edition, McGraw Hill education (India) Private Limited.
2. Discrete Mathematics, D.S. Malik & M.K. Sen, Revised edition Cengage Learning.
3. Elements of Discrete Mathematics, C. L. Liu and D. P. Mohapatra, 4th edition, McGraw Hill education (India) Private Limited.
4. Discrete Mathematics with Applications, Thomas Koshy, Elsevier.
5. Discrete and Combinatorial Mathematics, R. P. Grimaldi, Pearson.

CS304ES: DIGITAL LOGIC DESIGN

B.Tech. II Year I Sem.

L	T	P	C
3	0	0	3

Course Objectives:

- To understand basic number systems, codes and logical gates.
- To understand the concepts of Boolean algebra.
- To understand the use of minimization logic to solve the Boolean logic expressions..
- To understand the design of combinational and sequential circuits.
- To understand the state reduction methods for Sequential circuits.
- To understand the basics of various types of memories.

Course Outcomes:

- Able to understand number systems and codes.
- Able to solve Boolean expressions using Minimization methods.
- Able to design the sequential and combinational circuits.
- Able to apply state reduction methods to solve sequential circuits.

UNIT - I

Digital Systems, Binary Numbers, Number base conversions, Octal, Hexadecimal and other base numbers, complements, signed binary numbers, Floating point number representation, binary codes, Error detection and correction, binary storage and registers, binary logic, Boolean algebra and logic gates , Basic theorems and properties of Boolean Algebra, Boolean functions, canonical and standard forms, Digital Logic Gates.

UNIT - II

Gate–Level Minimization, The K-Map Method, Three-Variable Map, Four-Variable Map, Five-Variable Map , sum of products , product of sums simplification, Don't care conditions , NAND and NOR implementation and other two level implementations, Exclusive-OR function.

UNIT - III

Combinational Circuits (CC), Analysis procedure, Design Procedure, Combinational circuit for different code converters and other problems, Binary Adder-Subtractor, Decimal Adder, Binary Multiplier, Magnitude Comparator, Decoders, Encoders, Multiplexers, Demultiplexers.

UNIT - IV

Synchronous Sequential Circuits, Latches, Flip-flops, analysis of clocked sequential circuits, Registers, Shift registers, Ripple counters, Synchronous counters, other counters. Asynchronous Sequential Circuits -Introduction, Analysis procedure, Circuits with latches, Design procedure, Reduction of state and follow tables, Race- free state assignment, Hazards.

UNIT - V

Memory: Introduction, Random-Access memory, Memory decoding, ROM, Programmable Logic Array, Programmable Array Logic, Sequential programmable devices.

Register Transfer and Microoperations - Register Transfer Language, Register Transfer, Bus and Memory Transfers,

Arithmetic Microoperations, Logic Microoperations, Shift Microoperations, Arithmetic Logic Shift Unit.

TEXT BOOKS:

1. Digital Design, M. Morris Mano, M.D.Ciletti, 5th edition, Pearson.(Units I, II, III, IV, Part of Unit V)
2. Computer System Architecture, M.Morris Mano, 3rd edition, Pearson.(Part of Unit V)

REFERENCE BOOKS:

1. Switching and Finite Automata Theory, Z. Kohavi, Tata McGraw Hill.
2. Fundamentals of Logic Design, C. H. Roth, L. L. Kinney, 7th edition, Cengage Learning.
3. Fundamentals of Digital Logic & Micro Computer Design, 5TH Edition, M. Rafiquzzaman, John Wiley.

CS305ES: OBJECT ORIENTED PROGRAMMING THROUGH JAVA

B.Tech. II Year I Sem.

L	T	P	C
3	0	0	3

Course Objectives:

- To introduce the object oriented programming concepts.
- To understand object oriented programming concepts, and apply them in solving problems.
- To introduce the principles of inheritance and polymorphism; and demonstrate how they relate to the design of abstract classes
- To introduce the implementation of packages and interfaces
- To introduce the concepts of exception handling and multithreading.
- To introduce the design of Graphical User Interface using applets and swing controls.

Course Outcomes

- Able to solve real world problems using OOP techniques.
- Able to understand the use of abstract classes.
- Able to solve problems using java collection framework and I/o classes.
- Able to develop multithreaded applications with synchronization.
- Able to develop applets for web applications.
- Able to design GUI based applications

UNIT - I

Object-oriented thinking- A way of viewing world – Agents and Communities, messages and methods, Responsibilities, Classes and Instances, Class Hierarchies- Inheritance, Method binding, Overriding and Exceptions, Summary of Object-Oriented concepts. Java buzzwords, An Overview of Java, Data types, Variables and Arrays, operators, expressions, control statements, Introducing classes, Methods and Classes, String handling.

Inheritance– Inheritance concept, Inheritance basics, Member access, Constructors, Creating Multilevel hierarchy, super uses, using final with inheritance, Polymorphism-ad hoc polymorphism, pure polymorphism, method overriding, abstract classes, Object class, forms of inheritance- specialization, specification, construction, extension, limitation, combination, benefits of inheritance, costs of inheritance.

UNIT - II

Packages- Defining a Package, CLASSPATH, Access protection, importing packages.

Interfaces- defining an interface, implementing interfaces, Nested interfaces, applying interfaces, variables in interfaces and extending interfaces.

Stream based I/O(java.io) – The Stream classes-Byte streams and Character streams, Reading console Input and Writing Console Output, File class, Reading and writing Files, Random access file operations, The Console class, Serialization, Enumerations, auto boxing, generics.

UNIT - III

Exception handling - Fundamentals of exception handling, Exception types, Termination or resumptive models, Uncaught exceptions, using try and catch, multiple catch clauses, nested try statements, throw, throws and finally, built- in exceptions, creating own exception sub classes.

Multithreading- Differences between thread-based multitasking and process-based multitasking, Java thread model, creating threads, thread priorities, synchronizing threads, inter thread communication.

UNIT - IV

The Collections Framework (java.util)- Collections overview, Collection Interfaces, The Collection classes- Array List, Linked List, Hash Set, Tree Set, Priority Queue, Array Deque. Accessing a Collection via an Iterator, Using an Iterator, The For-Each alternative, Map Interfaces and Classes, Comparators, Collection algorithms, Arrays, The Legacy Classes and Interfaces- Dictionary, Hashtable ,Properties, Stack, Vector
More Utility classes, String Tokenizer, Bit Set, Date, Calendar, Random, Formatter, Scanner

UNIT - V

GUI Programming with Swing – Introduction, limitations of AWT, MVC architecture, components, containers. Understanding Layout Managers, Flow Layout, Border Layout, Grid Layout, Card Layout, Grid Bag Layout.

Event Handling- The Delegation event model- Events, Event sources, Event Listeners, Event classes, Handling mouse and keyboard events, Adapter classes, Inner classes, Anonymous Inner classes.

A Simple Swing Application, **Applets** – Applets and HTML, Security Issues, Applets and Applications, passing parameters to applets. Creating a Swing Applet, Painting in Swing, A Paint example, Exploring Swing Controls- JLabel and Image Icon, JText Field, The Swing Buttons- JButton, JToggle Button, JCheck Box, JRadio Button, JTabbed Pane, JScroll Pane, JList, JCombo Box, Swing Menus, Dialogs.

TEXT BOOKS

1. Java The complete reference, 9th edition, Herbert Schildt, McGraw Hill Education (India) Pvt. Ltd.
2. Understanding Object-Oriented Programming with Java, updated edition, T. Budd, Pearson Education.

REFERENCE BOOKS

1. An Introduction to programming and OO design using Java, J. Nino and F.A. Hosch, John Wiley & sons.
2. Introduction to Java programming, Y. Daniel Liang, Pearson Education.
3. Object Oriented Programming through Java, P. Radha Krishna, Universities Press.
4. Programming in Java, S. Malhotra, S. Chudhary, 2nd edition, Oxford Univ. Press.
5. Java Programming and Object oriented Application Development, R. A. Johnson, Cengage Learning.

CS306ES: DATA STRUCTURES THROUGH C++ LAB

B.Tech. II Year I Sem.

L	T	P	C
0	0	3	2

Course Objectives:

- To write and execute programs in C++ to solve problems using data structures such as arrays, linked lists, stacks, queues, trees, graphs, hash tables and search trees.
- To learn to write C++ programs to implement various sorting and searching algorithms

Course Outcomes:

- Able to identify the appropriate data structures and algorithms for solving real world problems.
- Able to implement various kinds of searching and sorting techniques.
- Able to implement data structures such as stacks, queues, Search trees, and hash tables to solve various computing problems.

1. Write a C++ program that uses functions to perform the following:
 - a) Create a singly linked list of integers.
 - b) Delete a given integer from the above linked list.
 - c) Display the contents of the above list after deletion.

2. Write a template based C++ program that uses functions to perform the following:
 - a) Create a doubly linked list of elements.
 - b) Delete a given element from the above doubly linked list.
 - c) Display the contents of the above list after deletion.

3. Write a C++ program that uses stack operations to convert a given infix expression into its postfix equivalent, Implement the stack using an array.

4. Write a C++ program to implement a double ended queue ADT using an array, using a doubly linked list.

5. Write a C++ program that uses functions to perform the following:
 - a) Create a binary search tree of characters.
 - b) Traverse the above Binary search tree recursively in preorder, in order and post order,

6. Write a C++ program that uses function templates to perform the following:
 - a) Search for a key element in a list of elements using linear search.
 - b) Search for a key element in a list of sorted elements using binary search.

7. Write a C++ program that implements Insertion sort algorithm to arrange a list of integers in ascending order.

8. Write a template based C++ program that implements selection sort algorithm to arrange a list of elements in descending order.
9. Write a template based C++ program that implements Quick sort algorithm to arrange a list of elements in ascending order.
10. Write a C++ program that implements Heap sort algorithm for sorting a list of integers in ascending order.
11. Write a C++ program that implements Merge sort algorithm for sorting a list of integers in ascending order
12. Write a C++ program to implement all the functions of a dictionary (ADT) using hashing.
13. Write a C++ program that implements Radix sort algorithm for sorting a list of integers in ascending order
14. Write a C++ program that uses functions to perform the following:
 - a) Create a binary search tree of integers.
 - b) Traverse the above Binary search tree non recursively in inorder.
15. Write a C++ program that uses functions to perform the following:
 - a) Create a binary search tree of integers.
 - b) Search for an integer key in the above binary search tree non recursively.
 - c) Search for an integer key in the above binary search tree recursively.

REFERENCE BOOKS

1. Data Structures using C++, D. S. Malik, 2nd edition, Cengage learning.
2. Data Structures using C++, V. Patil, Oxford University Press.
3. Fundamentals of Data structures in C++, 2nd edition, E. Horowitz, S. Sahni and D. Mehta, Universities Press.
4. C++ Plus Data Structures, 4th edition, Nell Dale, Jones and Bartlett student edition.

CS307ES: IT WORKSHOP

B.Tech. II Year I Sem.

L T P C
0 0 3 2

Course Objectives:

- The IT Workshop is a training lab course to get training on PC Hardware, Internet & World Wide Web, and Productivity tools for documentation, Spreadsheet computations, and Presentation.
- To introduce to a personal computer and its basic peripherals, the process of assembling a personal computer, installation of system software like MS Windows, Linux and the required device drivers, hardware and software level troubleshooting process.
- To introduce connecting the PC on to the internet from home and workplace and effectively usage of the internet, Usage of web browsers, email, newsgroups and discussion forums. To get knowledge in awareness of cyber hygiene, i.e., protecting the personal computer from getting infected with the viruses, worms and other cyber attacks.
- To introduce the usage of Productivity tools in crafting professional word documents, excel spreadsheets and power point presentations using open office tools and LaTeX.

Course Outcomes:

- Apply knowledge for computer assembling and software installation.
- Ability how to solve the trouble shooting problems.
- Apply the tools for preparation of PPT, Documentation and budget sheet etc.

PC Hardware: The students should work on working PC to disassemble and assemble to working condition and install operating system like Linux or any other on the same PC. Students are suggested to work similar tasks in the Laptop scenario wherever possible.

Problem 1: Every student should identify the peripherals of a computer, components in a CPU and its functions. Draw the block diagram of the CPU along with the configuration of each peripheral and submit to your instructor. Every student should disassemble and assemble the PC back to working condition.

Problem 2: Every student should individually install operating system like Linux or MS windows on the personal computer. The system should be configured as dual boot with both windows and Linux.

Problem 3: Hardware Troubleshooting: Students have to be given a PC which does not boot due to improper assembly or defective peripherals. They should identify the problem and fix it to get the computer back to working condition.

Problem 4: Software Troubleshooting: Students have to be given a malfunctioning CPU due to system software problems. They should identify the problem and fix it to get the computer back to working condition.

Internet & World Wide Web.

Problem 5: Orientation & Connectivity Boot Camp: Students should get connected to their Local Area Network and access the Internet. In the process they configure the TCP/IP setting. Finally students should demonstrate how to access the websites and email.

Problem 6: Web Browsers, Surfing the Web: Students customize their web browsers with the LAN proxy settings, bookmarks, search toolbars and pop up blockers. Also, plug-ins like Macromedia Flash and JRE for applets should be configured.

Problem 7: Search Engines & Netiquette: Students should know what search engines are and how to use the search engines. Usage of search engines like Google, Yahoo, ask.com and others should be demonstrated by student.

Problem 8: Cyber Hygiene: Students should learn about viruses on the internet and install antivirus software. Student should learn to customize the browsers to block pop ups, block active x downloads to avoid viruses and/or worms.

Problem 9: Develop home page: Student should learn to develop his/her home page using HTML consisting of his/her photo, name, address and education details as a table and his/her skill set as a list.

Productivity tools: LaTeX and Word

Word Orientation: An overview of LaTeX and Microsoft (MS) office / equivalent (FOSS) tool word should be learned: Importance of LaTeX and MS office / equivalent (FOSS) tool Word as word Processors, Details of the three tasks and features that should be covered in each, using LaTeX and word – Accessing, overview of toolbars, saving files, Using help and resources, rulers, format painter.

Problem 10: Using LaTeX and Word to create project certificate. Features to be covered:- Formatting Fonts in word, Drop Cap in word, Applying Text effects, Using Character Spacing, Borders and Colors, Inserting Header and Footer, Using Date and Time option in both LaTeX and Word.

Problem 11: Creating project abstract Features to be covered:-Formatting Styles, Inserting table, Bullets and Numbering, Changing Text Direction, Cell alignment, Footnote, Hyperlink, Symbols, Spell Check, Track Changes.

Problem 12: Creating a Newsletter: Features to be covered:- Table of Content, Newspaper columns, Images from files and clipart, Drawing toolbar and Word Art, Formatting Images, Textboxes, Paragraphs in word.

Problem 13 - Spreadsheet Orientation: Accessing, overview of toolbars, saving spreadsheet files, Using help and resources. **Creating a Scheduler:-** Gridlines, Format Cells, Summation, auto fill, Formatting Text

Problem 14: Calculating GPA - .Features to be covered:- Cell Referencing, Formulae in spreadsheet – average, std. deviation, Charts, Renaming and Inserting worksheets, Hyper linking, Count function, Sorting, Conditional formatting.

Problem 15: Creating Power Point: Student should work on basic power point utilities and tools in Latex and Ms Office/equivalent (FOSS) which help them create basic power point presentation. PPT Orientation, Slide Layouts, Inserting Text, Formatting Text, Bullets and Numbering, Auto Shapes, Lines and Arrows, Hyperlinks, Inserting Images, Tables and Charts

REFERENCE BOOKS:

1. Introduction to Information Technology, IITL Education Solutions limited, Pearson Education.
2. LaTeX Companion – Leslie Lamport, PHI/Pearson.
3. Comdex Information Technology course tool kit Vikas Gupta, WILEY Dreamtech
4. IT Essentials PC Hardware and Software Companion Guide Third Edition by David Anfinson and Ken Quamme. – CISCO Press, Pearson Education.
5. PC Hardware and A+ Handbook – Kate J. Chase PHI (Microsoft)

CS308ES: OBJECT ORIENTED PROGRAMMING THROUGH JAVA LAB

B.Tech. II Year I Sem.

L	T	P	C
0	0	3	2

Course Objectives:

- To write programs using abstract classes.
- To write programs for solving real world problems using java collection frame work.
- To write multithreaded programs.
- To write GUI programs using swing controls in Java.
- To introduce java compiler and eclipse platform.
- To impart hands on experience with java programming.

Course Outcomes:

- Able to write programs for solving real world problems using java collection frame work.
- Able to write programs using abstract classes.
- Able to write multithreaded programs.
- Able to write GUI programs using swing controls in Java.

Note:

1. Use Linux and MySQL for the Lab Experiments. Though not mandatory, encourage the use of Eclipse platform.
2. The list suggests the minimum program set. Hence, the concerned staff is requested to add more problems to the list as needed.
1. Use Eclipse or Net bean platform and acquaint with the various menus. Create a test project, add a test class, and run it. See how you can use auto suggestions, auto fill. Try code formatter and code refactoring like renaming variables, methods, and classes. Try debug step by step with a small program of about 10 to 15 lines which contains at least one if else condition and a for loop.
2. Write a Java program that works as a simple calculator. Use a grid layout to arrange buttons for the digits and for the +, -,*, % operations. Add a text field to display the result. Handle any possible exceptions like divided by zero.
3. A) Develop an applet in Java that displays a simple message.
b) Develop an applet in Java that receives an integer in one text field, and computes its factorial Value and returns it in another text field, when the button named “Compute” is clicked.
4. Write a Java program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields, Num1 and Num2. The division of Num1

and Num 2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer, the program would throw a Number Format Exception. If Num2 were Zero, the program would throw an Arithmetic Exception. Display the exception in a message dialog box.

5. Write a Java program that implements a multi-thread application that has three threads. First thread generates random integer every 1 second and if the value is even, second thread computes the square of the number and prints. If the value is odd, the third thread will print the value of cube of the number.
6. Write a Java program for the following:
 - i) Create a doubly linked list of elements.
 - ii) Delete a given element from the above list.
 - iii) Display the contents of the list after deletion.
7. Write a Java program that simulates a traffic light. The program lets the user select one of three lights: red, yellow, or green with radio buttons. On selecting a button, an appropriate message with “Stop” or “Ready” or “Go” should appear above the buttons in selected color. Initially, there is no message shown.
8. Write a Java program to create an abstract class named Shape that contains two integers and an empty method named print Area (). Provide three classes named Rectangle, Triangle, and Circle such that each one of the classes extends the class Shape. Each one of the classes contains only the method print Area () that prints the area of the given shape.
9. Suppose that a table named Table.txt is stored in a text file. The first line in the file is the header, and the remaining lines correspond to rows in the table. The elements are separated by commas. Write a java program to display the table using Labels in Grid Layout.
10. Write a Java program that handles all mouse events and shows the event name at the center of the window when a mouse event is fired (Use Adapter classes).
11. Write a Java program that loads names and phone numbers from a text file where the data is organized as one line per record and each field in a record are separated by a tab (t). It takes a name or phone number as input and prints the corresponding other value from the hash table (hint: use hash tables).
12. Write a Java program that correctly implements the producer – consumer problem using the concept of interthread communication.
13. Write a Java program to list all the files in a directory including the files present in all its subdirectories.

14. Write a Java program that implements Quick sort algorithm for sorting a list of names in ascending order

15. Write a Java program that implements Bubble sort algorithm for sorting in descending order and also shows the number of interchanges occurred for the given set of integers.

REFERENCE BOOKS

1. Java for Programmers, P. J. Deitel and H. M. Deitel, 10th Edition Pearson education.
2. Thinking in Java, Bruce Eckel, Pearson Education.
3. Java Programming, D. S. Malik and P. S. Nair, Cengage Learning.
4. Core Java, Volume 1, 9th edition, Cay S. Horstmann and G Cornell, Pearson.

MC300ES: ENVIRONMENTAL SCIENCE AND TECHNOLOGY

B.Tech. II Year I Sem.

L	T	P	C
3	0	0	0

Course Objectives:

- Understanding the importance of ecological balance for sustainable development.
- Understanding the impacts of developmental activities and mitigation measures
- Understanding the environmental policies and regulations

Course Outcomes:

Based on this course, the Engineering graduate will understand /evaluate / develop technologies on the basis of ecological principles and environmental regulations which in turn helps in sustainable development

UNIT - I

Ecosystems: Definition, Scope and Importance of ecosystem. Classification, structure, and function of an ecosystem, Food chains, food webs, and ecological pyramids. Flow of energy, Biogeochemical cycles, Bioaccumulation, Biomagnification, ecosystem value, services and carrying capacity, Field visits.

UNIT - II

Natural Resources: Classification of Resources: Living and Non-Living resources, **water resources:** use and over utilization of surface and ground water, floods and droughts, Dams: benefits and problems. **Mineral resources:** use and exploitation, environmental effects of extracting and using mineral resources, **Land resources:** Forest resources, **Energy resources:** growing energy needs, renewable and non renewable energy sources, use of alternate energy source, case studies.

UNIT - III

Biodiversity And Biotic Resources: Introduction, Definition, genetic, species and ecosystem diversity. Value of biodiversity; consumptive use, productive use, social, ethical, aesthetic and optional values. India as a mega diversity nation, Hot spots of biodiversity. Field visit. Threats to biodiversity: habitat loss, poaching of wildlife, man-wildlife conflicts; conservation of biodiversity: In-Situ and Ex-situ conservation. National Biodiversity act.

UNIT - IV

Environmental Pollution and Control Technologies: Environmental Pollution: Classification of pollution, **Air Pollution:** Primary and secondary pollutants, Automobile and Industrial pollution, Ambient air quality standards. **Water pollution:** Sources and types of pollution, drinking water quality standards. **Soil Pollution:** Sources and types, Impacts of modern agriculture, degradation of soil. **Noise Pollution:** Sources and Health hazards, standards, **Solid waste:** Municipal Solid Waste management, composition and characteristics

of e-Waste and its management. **Pollution control technologies:** Wastewater Treatment methods: Primary, secondary and Tertiary.

Overview of air pollution control technologies, Concepts of bioremediation. **Global Environmental Problems and Global Efforts:** Climate change and impacts on human environment. Ozone depletion and Ozone depleting substances (ODS). Deforestation and desertification. International conventions / Protocols: Earth summit, Kyoto protocol, and Montréal Protocol.

UNIT - V

Environmental Policy, Legislation & EIA: Environmental Protection act, Legal aspects Air Act- 1981, Water Act, Forest Act, Wild life Act, Municipal solid waste management and handling rules, biomedical waste management and handling rules, hazardous waste management and handling rules. EIA: EIA structure, methods of baseline data acquisition. Overview on Impacts of air, water, biological and Socio-economical aspects. Strategies for risk assessment, Concepts of Environmental Management Plan (EMP). **Towards Sustainable Future:** Concept of Sustainable Development, Population and its explosion, Crazy Consumerism, Environmental Education, Urban Sprawl, Human health, Environmental Ethics, Concept of Green Building, Ecological Foot Print, Life Cycle assessment (LCA), Low carbon life style.

TEXT BOOKS:

1. Textbook of Environmental Studies for Undergraduate Courses by Erach Bharucha for University Grants Commission.
2. Environmental Studies by R. Rajagopalan, Oxford University Press.

REFERENCE BOOKS:

1. Environmental Science: towards a sustainable future by Richard T. Wright. 2008 PHL Learning Private Ltd. New Delhi.
2. Environmental Engineering and science by Gilbert M. Masters and Wendell P. Ela. 2008 PHI Learning Pvt. Ltd.
3. Environmental Science by Daniel B. Botkin & Edward A. Keller, Wiley INDIA edition.
4. Environmental Studies by Anubha Kaushik, 4th Edition, New age international publishers.
5. Text book of Environmental Science and Technology - Dr. M. Anji Reddy 2007, BS Publications.

CS401ES: COMPUTER ORGANIZATION

B.Tech. II Year II Sem.

L	T	P	C
4	0	0	4

Course Objectives:

- To understand basic components of computers.
- To understand the architecture of 8086 processor.
- To understand the instruction sets, instruction formats and various addressing modes of 8086.
- To understand the representation of data at the machine level and how computations are performed at machine level.
- To understand the memory organization and I/O organization.
- To understand the parallelism both in terms of single and multiple processors.

Course Outcomes:

- Able to understand the basic components and the design of CPU, ALU and Control Unit.
- Ability to understand memory hierarchy and its impact on computer cost/performance.
- Ability to understand the advantage of instruction level parallelism and pipelining for high performance Processor design.
- Ability to understand the instruction set, instruction formats and addressing modes of 8086.
- Ability to write assembly language programs to solve problems.

UNIT - I

Digital Computers: Introduction, Block diagram of Digital Computer, Definition of Computer Organization, Computer Design and Computer Architecture.

Basic Computer Organization and Design: Instruction codes, Computer Registers, Computer instructions, Timing and Control, Instruction cycle, Memory Reference Instructions, Input – Output and Interrupt, Complete Computer Description.

Micro Programmed Control: Control memory, Address sequencing, micro program example, design of control unit.

UNIT - II

Central Processing Unit: The 8086 Processor Architecture, Register organization, Physical memory organization, General Bus Operation, I/O Addressing Capability, Special Processor Activities, Minimum and Maximum mode system and timings.

8086 Instruction Set and Assembler Directives-Machine language instruction formats, Addressing modes, Instruction set of 8086, Assembler directives and operators.

UNIT - III

Assembly Language Programming with 8086- Machine level programs, Machine coding the programs, Programming with an assembler, Assembly Language example programs.
Stack structure of 8086, Interrupts and Interrupt service routines, Interrupt cycle of 8086, Interrupt programming, Passing parameters to procedures, Macros, Timings and Delays.

UNIT - IV

Computer Arithmetic: Introduction, Addition and Subtraction, Multiplication Algorithms, Division Algorithms, Floating - point Arithmetic operations.

Input-Output Organization: Peripheral Devices, Input-Output Interface, Asynchronous data transfer, Modes of Transfer, Priority Interrupt, Direct memory Access, Input –Output Processor (IOP), Intel 8089 IOP.

UNIT - V

Memory Organization: Memory Hierarchy, Main Memory, Auxiliary memory, Associate Memory, Cache Memory.

Pipeline and Vector Processing: Parallel Processing, Pipelining, Arithmetic Pipeline, Instruction Pipeline, RISC Pipeline, Vector Processing, Array Processors.

Multi Processors: Characteristics of Multiprocessors, Interconnection Structures, Inter processor arbitration, Inter processor communication, and synchronization.

TEXT BOOKS:

1. Computer System Architecture, M. Moris Mano, Third Edition, Pearson. (**UNIST-I, IV, V**)
2. Advanced Microprocessors and Peripherals, K M Bhurchandi, A.K Ray ,3rd edition, McGraw Hill India Education Private Ltd. (**UNITS - II, III**).

REFERENCE:

1. Microprocessors and Interfacing, D V Hall, SSSP Rao, 3rd edition, McGraw Hill India Education Private Ltd.
2. Carl Hamacher, Zvonko Vranesic, Safwat Zaky: Computer Organization, 5th Edition, Tata McGraw Hill, 2002
3. Computer Organization and Architecture, William Stallings, 9th Edition, Pearson.
4. David A. Patterson, John L. Hennessy: Computer Organization and Design – The Hardware / Software Interface ARM Edition, 4th Edition, Elsevier, 2009.

CS402ES: DATABASE MANAGEMENT SYSTEMS

B.Tech. II Year II Sem.

L	T	P	C
4	0	0	4

Course Objectives:

- To understand the basic concepts and the applications of database systems.
- To master the basics of SQL and construct queries using SQL.
- To understand the relational database design principles.
- To become familiar with the basic issues of transaction processing and concurrency control.
- To become familiar with database storage structures and access techniques.

Course Outcomes:

- Demonstrate the basic elements of a relational database management system.
- Ability to identify the data models for relevant problems.
- Ability to design entity relationship model and convert entity relationship diagrams into RDBMS and formulate SQL queries on the data.
- Apply normalization for the development of application software.

UNIT - I

Introduction: Database System Applications, Purpose of Database Systems, View of Data, Database Languages – DDL, DML, Relational Databases, Database Design, Data Storage and Querying, Transaction Management, Database Architecture, Data Mining and Information Retrieval, Specialty Databases, Database Users and Administrators, History of Database Systems.

Introduction to Data base design: Database Design and ER diagrams, Entities, Attributes and Entity sets, Relationships and Relationship sets, Additional features of ER Model, Conceptual Design with the ER Model, Conceptual Design for Large enterprises.

Relational Model: Introduction to the Relational Model, Integrity Constraints over Relations, Enforcing Integrity constraints, Querying relational data, Logical data base Design: ER to Relational, Introduction to Views, Destroying /Altering Tables and Views.

UNIT - II

Relational Algebra and Calculus: Preliminaries, Relational Algebra, Relational calculus – Tuple relational Calculus, Domain relational calculus, Expressive Power of Algebra and calculus.

SQL: Queries, Constraints, Triggers: Form of Basic SQL Query, UNION, INTERSECT, and EXCEPT, Nested Queries, Aggregate Operators, NULL values Complex Integrity Constraints in SQL, Triggers and Active Data bases, Designing Active Databases..

UNIT - III

Schema Refinement and Normal Forms: Introduction to Schema Refinement, Functional Dependencies - Reasoning about FDs, Normal Forms, Properties of Decompositions, Normalization, Schema Refinement in Database Design, Other Kinds of Dependencies.

UNIT - IV

Transaction Management: Transactions, Transaction Concept, A Simple Transaction Model, Storage Structure, Transaction Atomicity and Durability, Transaction Isolation, Serializability, Transaction Isolation and Atomicity Transaction Isolation Levels, Implementation of Isolation Levels.

Concurrency Control: Lock-Based Protocols, Multiple Granularity, Timestamp-Based Protocols, Validation-Based Protocols, Multiversion Schemes.

Recovery System-Failure Classification, Storage, Recovery and Atomicity, Recovery Algorithm, Buffer Management, Failure with loss of nonvolatile storage, Early Lock Release and Logical Undo Operations, Remote Backup systems.

UNIT - V

Storage and Indexing: Overview of Storage and Indexing: Data on External Storage, File Organization and Indexing, Index Data Structures, Comparison of File Organizations.

Tree-Structured Indexing: Intuition for tree Indexes, Indexed Sequential Access Method (ISAM), B+ Trees: A Dynamic Index Structure, Search, Insert, Delete.

Hash- Based Indexing: Static Hashing, Extendible hashing, Linear Hashing, Extendible vs. Linear Hashing.

TEXT BOOKS:

1. Data base Management Systems, Raghu Ramakrishnan, Johannes Gehrke, McGraw Hill Education (India) Private Limited, 3rd Edition. (**Part of UNIT-I, UNIT-II, UNIT-III, UNIT-V**)
2. Data base System Concepts, A. Silberschatz, Henry. F. Korth, S. Sudarshan, McGraw Hill Education(India) Private Limited 1, 6th edition.(**Part of UNIT-I, UNIT-IV**)

REFERENCE BOOKS:

1. Database Systems, 6th edition, R Elmasri, Shamkant B.Navathe, Pearson Education.
2. Database System Concepts, Peter Rob & Carlos Coronel, Cengage Learning.
3. Introduction to Database Management, M. L. Gillenson and others, Wiley Student Edition.
4. Database Development and Management, Lee Chao, Auerbach publications, Taylor & Francis Group.
5. Introduction to Database Systems, C. J. Date, Pearson Education.

CS403ES: OPERATING SYSTEMS

B.Tech. II Year II Sem.

L	T	P	C
3	0	0	3

Course Objectives:

- To understand the OS role in the overall computer system
- To study the operations performed by OS as a resource manager
- To understand the scheduling policies of OS
- To understand the different memory management techniques
- To understand process concurrency and synchronization
- To understand the concepts of input/output, storage and file management
- To understand the goals and principles of protection
- Introduce system call interface for file and process management
- To study different OS and compare their features.

Course Outcomes:

- Apply optimization techniques for the improvement of system performance.
- Ability to design and solve synchronization problems.
- Learn about minimization of turnaround time, waiting time and response time and also maximization of throughput by keeping CPU as busy as possible.
- Ability to change access controls to protect files.
- Ability to compare the different operating systems.

UNIT - I

Overview-Introduction-Operating system objectives, User view, System view, Operating system definition ,Computer System Organization, Computer System Architecture, OS Structure, OS Operations, Process Management, Memory Management, Storage Management, Protection and Security, Computing Environments.

Operating System services, User and OS Interface, System Calls, Types of System Calls, System Programs, Operating System Design and Implementation, OS Structure.

UNIT - II

Process and CPU Scheduling - Process concepts-The Process, Process State, Process Control Block, Threads, Process Scheduling-Scheduling Queues, Schedulers, Context Switch, Operations on Processes, System calls-fork(),exec(),wait(),exit(), Interprocess communication-ordinary pipes and named pipes in Unix.

Process Scheduling-Basic concepts, Scheduling Criteria, Scheduling algorithms, Multiple-Processor Scheduling, Real-Time Scheduling, Thread scheduling, Linux scheduling and Windows scheduling.

Process Synchronization, Background, The Critical Section Problem, Peterson's solution, Synchronization Hardware, Semaphores, Classic Problems of Synchronization, Monitors, Synchronization in Linux and Windows.

UNIT - III

Memory Management and Virtual Memory – Memory Management Strategies- Background, Swapping, Contiguous Memory Allocation, Segmentation, Paging, Structure of Page Table, IA-32 Segmentation, IA-32 Paging.

Virtual Memory Management-Background, Demand Paging, Copy-on-Write, Page Replacement, Page Replacement Algorithms, Allocation of Frames, Thrashing, Virtual memory in Windows..

UNIT - IV

Storage Management-File System- Concept of a File, System calls for file operations - open (), read (), write (), close (), seek (), unlink (), Access methods, Directory and Disk Structure, File System Mounting, File Sharing, Protection.

File System Implementation - File System Structure, File System Implementation, Directory Implementation, Allocation methods, Free-space Management, Efficiency, and Performance.

Mass Storage Structure – Overview of Mass Storage Structure, Disk Structure, Disk Attachment, Disk Scheduling, Disk Management, Swap space Management

UNIT - V

Deadlocks - System Model, Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, and Recovery from Deadlock.

Protection – System Protection, Goals of Protection, Principles of Protection, Domain of Protection, Access Matrix, Implementation of Access Matrix, Access Control, Revocation of Access Rights, Capability-Based Systems, Language-Based Protection.

TEXT BOOKS:

1. Operating System Concepts , Abraham Silberschatz, Peter B. Galvin, Greg Gagne, 9th Edition, Wiley, 2016 India Edition
2. Operating Systems – Internals and Design Principles, W. Stallings, 7th Edition, Pearson.

REFERENCE BOOKS:

1. Modern Operating Systems, Andrew S Tanenbaum, 3rd Edition, PHI
2. Operating Systems: A concept-based Approach, 2nd Edition, D.M. Dhamdhare, TMH.
3. Principles of Operating Systems, B. L. Stuart, Cengage learning, India Edition.
4. An Introduction to Operating Systems, P.C.P. Bhatt, PHI.
5. Principles of Operating systems, Naresh Chauhan, Oxford University Press.

CS404ES: FORMAL LANGUAGES AND AUTOMATA THEORY

B.Tech. II Year II Sem.

L	T	P	C
3	0	0	3

Course Objectives:

- To provide introduction to some of the central ideas of theoretical computer science from the perspective of formal languages.
- To introduce the fundamental concepts of formal languages, grammars and automata theory.
- Classify machines by their power to recognize languages.
- Employ finite state machines to solve problems in computing.
- To understand deterministic and non-deterministic machines.
- To understand the differences between decidability and undecidability.

Course Outcomes:

- Able to understand the concept of abstract machines and their power to recognize the languages.
- Able to employ finite state machines for modeling and solving computing problems.
- Able to design context free grammars for formal languages.
- Able to distinguish between decidability and undecidability.
- Able to gain proficiency with mathematical tools and formal methods.

UNIT - I

Introduction to Finite Automata, Structural Representations, Automata and Complexity, the Central Concepts of Automata Theory – Alphabets, Strings, Languages, Problems. Deterministic Finite Automata, Nondeterministic Finite Automata, an application: Text Search, Finite Automata with Epsilon-Transitions.

UNIT - II

Regular Expressions, Finite Automata and Regular Expressions, Applications of Regular Expressions, Algebraic Laws for Regular Expressions, Properties of Regular Languages- Pumping Lemma for Regular Languages, Applications of the Pumping Lemma, Closure Properties of Regular Languages, Decision Properties of Regular Languages, Equivalence and Minimization of Automata.

UNIT - III

Context-Free Grammars: Definition of Context-Free Grammars, Derivations Using a Grammar, Leftmost and Rightmost Derivations, the Language of a Grammar, Sentential Forms, Parse Trees, Applications of Context-Free Grammars, Ambiguity in Grammars and Languages.

Push Down Automata,: Definition of the Pushdown Automaton, the Languages of a PDA, Equivalence of PDA's and CFG's, Deterministic Pushdown Automata.

UNIT - IV

Normal Forms for Context- Free Grammars, the Pumping Lemma for Context-Free Languages, Closure Properties of Context-Free Languages. Decision Properties of CFL's - Complexity of Converting among CFG's and PDA's, Running time of conversions to Chomsky Normal Form.

Introduction to Turing Machines-Problems That Computers Cannot Solve, The Turing Machine, Programming Techniques for Turing Machines, Extensions to the basic Turing machine, Restricted Turing Machines, Turing Machines, and Computers

UNIT - V

Undecidability: A Language that is Not Recursively Enumerable, An Undecidable Problem That is RE, Undecidable Problems about Turing Machines, Post's Correspondence Problem, Other Undecidable Problems, Intractable Problems: The Classes P and NP, An NP-Complete Problem.

TEXT BOOKS:

1. Introduction to Automata Theory, Languages, and Computation, 3rd Edition, John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman, Pearson Education.
2. Introduction to the Theory of Computation, Michael Sipser, 3rd edition, Cengage Learning.

REFERENCE BOOKS:

1. Introduction to Languages and The Theory of Computation, John C Martin, TMH.
2. Introduction to Computer Theory, Daniel I.A. Cohen, John Wiley.
3. A Text book on Automata Theory, P. K. Srimani, Nasir S. F. B, Cambridge University Press.
4. Introduction to Formal languages Automata Theory and Computation Kamala Krithivasan, Rama R, Pearson.
5. Theory of Computer Science – Automata languages and computation, Mishra and Chandrashekar, 2nd edition, PHI.

SM405ES: BUSINESS ECONOMICS AND FINANCIAL ANALYSIS

B.Tech. II Year II Sem.

L	T	P	C
3	0	0	3

Course Objective: To learn the basic Business types, impact of the Economy on Business and Firms specifically. To analyze the Business from the Financial Perspective.

Course Outcome: The students will understand the various Forms of Business and the impact of economic variables on the Business. The Demand, Supply, Production, Cost, Market Structure, Pricing aspects are learnt. The Students can study the firm's financial position by analysing the Financial Statements of a Company.

UNIT – I

Introduction to Business and Economics:

Business: Structure of Business Firm, Theory of Firm, Types of Business Entities, Limited Liability Companies, Sources of Capital for a Company, Non-Conventional Sources of Finance.

Economics: Significance of Economics, Micro and Macro Economic Concepts, Concepts and Importance of National Income, Inflation, Money Supply in Inflation, Business Cycle, Features and Phases of Business Cycle. Nature and Scope of Business Economics, Role of Business Economist, Multidisciplinary nature of Business Economics.

UNIT – II

Demand and Supply Analysis:

Elasticity of Demand: Elasticity, Types of Elasticity, Law of Demand, Measurement and Significance of Elasticity of Demand, Factors affecting Elasticity of Demand, Elasticity of Demand in decision making, Demand Forecasting: Characteristics of Good Demand Forecasting, Steps in Demand Forecasting, Methods of Demand Forecasting.

Supply Analysis: Determinants of Supply, Supply Function & Law of Supply.

UNIT- III

Production, Cost, Market Structures & Pricing:

Production Analysis: Factors of Production, Production Function, Production Function with one variable input, two variable inputs, Returns to Scale, Different Types of Production Functions.

Cost analysis: Types of Costs, Short run and Long run Cost Functions.

Market Structures: Nature of Competition, Features of Perfect competition, Monopoly, Oligopoly, and Monopolistic Competition.

Pricing: Types of Pricing, Product Life Cycle based Pricing, Break Even Analysis, and Cost Volume Profit Analysis.

UNIT - IV

Financial Accounting: Accounting concepts and Conventions, Accounting Equation, Double-Entry system of Accounting, Rules for maintaining Books of Accounts, Journal, Posting to Ledger, Preparation of Trial Balance, Elements of Financial Statements, and Preparation of Final Accounts.

UNIT - V

Financial Analysis through Ratios: Concept of Ratio Analysis, Liquidity Ratios, Turnover Ratios, Profitability Ratios, Proprietary Ratios, Solvency, Leverage Ratios (simple problems). Introduction to Fund Flow and Cash Flow Analysis (simple problems).

TEXT BOOKS:

1. D. D. Chaturvedi, S. L. Gupta, Business Economics - Theory and Applications, International Book House Pvt. Ltd. 2013.
2. Dhanesh K Khatri, Financial Accounting, Tata McGraw Hill, 2011.
3. Geethika Ghosh, Piyali Gosh, Purba Roy Choudhury, Managerial Economics, 2e, Tata McGraw Hill Education Pvt. Ltd. 2012.

REFERENCES:

1. Paresh Shah, Financial Accounting for Management 2e, Oxford Press, 2015.
2. S. N. Maheshwari, Sunil K Maheshwari, Sharad K Maheshwari, Financial Accounting, 5e, Vikas Publications, 2013.

CS406ES: COMPUTER ORGANIZATION LAB

B.Tech. II Year II Sem.

L	T	P	C
0	0	3	2

Exercises in Digital Logic Design:

1. Implement Logic gates using NAND and NOR gates
2. Design a Full adder using gates
3. Design and implement the 4:1 MUX, 8:1 MUX using gates /ICs.
4. Design and Implement a 3 to 8 decoder using gates
5. Design a 4 bit comparator using gates/IC
6. Design and Implement a 4 bit shift register using Flip flops
7. Design and Implement a Decade counter

Exercises in Micro Processor programming:

Write assembly language programs for the following using GNU Assembler.

1. Write assembly language programs to evaluate the expressions:
 - i) $a = b + c - d * e$
 - ii) $z = x * y + w - v + u / k$
 - a. Considering 8-bit, 16 bit and 32 bit binary numbers as b, c, d, e.
 - b. Considering 2 digit, 4 digit and 8 digit BCD numbers.
Take the input in consecutive memory locations and also Display the results by using "int xx" of 8086. Validate program for the boundary conditions.
2. Write an ALP of 8086 to take N numbers as input. And do the following operations on them.
 - a. Arrange in ascending and descending order.
3. Write an ALP of 8086 to take N numbers as input. And do the following operations on them.
 - a. Find max and minimum
 - b. Find average
Considering 8-bit, 16 bit binary numbers and 2 digit, 4digit and 8 digit BCD numbers. Display the results by using "int xx" of 8086. Validate program for the boundary conditions.
4. Write an ALP of 8086 to take a string of as input (in 'C' format)and do the following Operations on it.
 - a. Find the length
 - b. Find it is Palindrome or n.
Considering 8-bit, 16 bit binary numbers and 2 digit, 4digit and 8 digit BCD numbers. Display the results by using "int xx" of 8086. Validate program for the boundary conditions.
5. Write an ALP of 8086 to take a string of as input (in 'C' format) and do the following Operations on it.
 - a. Find whether given string substring or not.

6. Write an ALP of 8086 to take a string of as input (in 'C' format) and do the following Operations on it
 - a. Find the Armstrong number
 - b. Find the Fibonacci series for n numbersDisplay the results by using "int xx" of 8086.
7. Write the ALP to implement the above operations as procedures and call from the main procedure.
8. Write an ALP of 8086 to find the factorial of a given number as a Procedure and call from the main program which display the result.

REFERENCE BOOKS:

1. Switching theory and logic design –A. Anand Kumar PHI, 2013
2. Advanced microprocessor & Peripherals-A. K. Ray and K. M. Bherchandavi, TMH, 2nd edition.
3. Switching and Finite Automatic theory-Zvi Kohavi, Niraj K.Jha Cambridge, 3rd edition
4. Digital Design –Morris Mano, PHI, 3rd edition
5. Microprocessor and Interfacing –Douglas V. Hall, TMGH 2nd edition.

CS407ES: DATABASE MANAGEMENT SYSTEMS LAB

B.Tech. II Year II Sem.

L	T	P	C
0	0	3	2

Course Objectives: This lab enables the students to practice the concepts learnt in the subject DBMS by developing a database for an example company named “Roadway Travels” whose description is as follows. The student is expected to practice the designing, developing and querying a database in the context of example database “Roadway travels”. Students are expected to use “Mysql” database.

Course Outcomes:

- Ability to design and implement a database schema for given problem.
- Apply the normalization techniques for development of application software to realistic problems.
- Ability to formulate queries using SQL DML/DDD/DCL commands.

Roadway Travels: "Roadway Travels" is in business since 1997 with several buses connecting different places in India. Its main office is located in Hyderabad.

The company wants to *computerize its operations* in the following areas:

- Reservations and Ticketing
- Cancellations

Reservations & Cancellation: Reservations are directly handled by booking office. Reservations can be made 30 days in advance and tickets issued to passenger. One Passenger/person can book many tickets (to his/her family). Cancellations are also directly handed at the booking office.

In the process of *computerization* of **Roadway Travels** you have to design and develop a Database which consists the data of Buses, Passengers, Tickets, and Reservation and cancellation details. You should also develop query's using SQL to retrieve the data from the database.

The above process involves many steps like 1. Analyzing the [problem](#) and identifying the Entities and Relationships, 2. E-R Model 3. Relational Model 4. Normalization 5. Creating the database 6. Querying. *Students are supposed to work on these steps week wise and finally create a complete “Database System” to Roadway Travels.* Examples are given at every experiment for guidance to students.

Experiment 1: E-R Model

Analyze the [problem](#) carefully and come up with the entities in it. Identify what data has to be persisted in the database. This contains the entities, attributes etc.

Identify the primary keys for all the entities. Identify the other keys like candidate keys, partial keys, if any.

Example:

Entities: 1. BUS 2. Ticket 3. Passenger

Relationships: 1. Reservation 2. Cancellation

Primary Key Attributes: 1. Ticket ID (Ticket Entity) 2. Passport ID (Passenger Entity)
3. Bus_NO (Bus Entity)

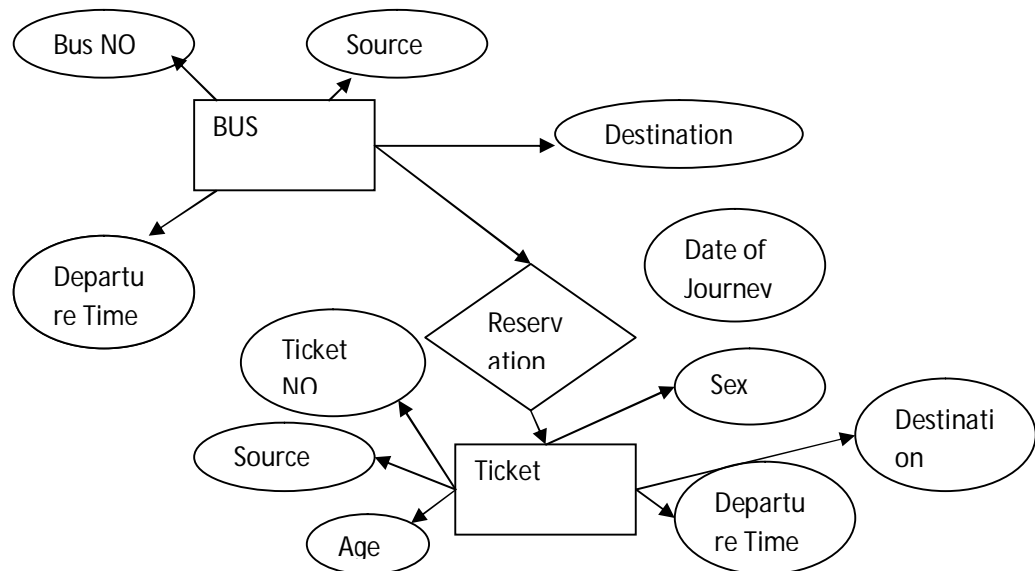
Apart from the above mentioned entities you can identify more. The above mentioned are few.

Note: The student is required to submit a document by writing the Entities and Keys to the lab teacher.

Experiment 2: Concept design with E-R Model

Relate the entities appropriately. Apply cardinalities for each relationship. Identify strong entities and weak entities (if any). Indicate the type of relationships (total / partial). Try to incorporate generalization, aggregation, specialization etc wherever required.

Example: E-R diagram for bus



Note: The student is required to submit a document by drawing the E-R Diagram to the lab teacher.

Experiment 3: Relational Model

Represent all the entities (Strong, Weak) in tabular fashion. Represent relationships in a tabular fashion. There are different ways of representing relationships as tables based on the cardinality. Represent attributes as columns in tables or as tables based on the requirement.

Different types of attributes (Composite, Multi-valued, and Derived) have different way of representation.

Example: The passenger tables look as below. This is an example. You can add more attributes based on your E-R model. This is not a normalized table.

Passenger

Name	Age	Sex	Address	<u>Passport ID</u>	Ticket _id

Note: The student is required to submit a document by Represent relationships in a tabular fashion to the lab teacher.

Experiment 4: Normalization

Database normalization is a technique for designing relational database tables to minimize duplication of information and, in so doing, to safeguard the database against certain types of logical or structural problems, namely data anomalies. For example, when multiple instances of a given piece of information occur in a table, the possibility exists that these instances will not be kept consistent when the data within the table is updated, leading to a loss of data integrity. A table that is sufficiently normalized is less vulnerable to problems of this kind, because its structure reflects the basic assumptions for when multiple instances of the same information should be represented by a single instance only.

For the above table in the First normalization we can remove the multi valued attribute Ticket_id and place it in another table along with the primary key of passenger.

First Normal Form: The above table can be divided into two tables as shown below.

Passenger

Name	Age	Sex	Address	<u>Passport ID</u>

<u>Passport ID</u>	Ticket_id

You can do the second and third normal forms if required. Any how Normalized tables are given at the end.

Experiment 5: Installation of Mysql and practicing DDL commands

Installation of MySQL. In this week you will learn Creating databases, How to create tables, altering the database, dropping tables and databases if not required. You will also try truncate, rename commands etc.

Example for creation of a normalized “Passenger” table.

```
CREATE TABLE Passenger (  
    Passport_id INTEGER PRIMARY KEY,  
    Name VARCHAR (50) Not NULL,  
    Age Integer Not NULL,  
    Sex Char,  
    Address VARCHAR (50) Not NULL);
```

Similarly create all other tables.

Note: Detailed creation of tables is given at the end.

Experiment 6: Practicing DML commands

DML commands are used to for managing data within schema objects. Some examples:

- SELECT - retrieve data from the a database
- INSERT - insert data into a table
- UPDATE - updates existing data within a table
- DELETE - deletes all records from a table, the space for the records remain

Inserting values into “Bus” table:

Insert into Bus values (1234,'hyderabad', 'tirupathi');

Insert into Bus values (2345,'hyderabad','Banglore');

Insert into Bus values (23,'hyderabad','Kolkata');

Insert into Bus values (45,'Tirupathi','Banglore');

Insert into Bus values (34,'hyderabad','Chennai');

Inserting values into “Passenger” table:

Insert into Passenger values (1, 45,'ramesh', 45,'M','abc123');

Insert into Passenger values (2, 78,'geetha', 36,'F','abc124');

Insert into Passenger values (45, 90,'ram', 30,'M','abc12');

Insert into Passenger values (67, 89,'ravi', 50,'M','abc14');

Insert into Passenger values (56, 22,'seetha', 32,'F','abc55');

Few more Examples of DML commands:

Select * from Bus; (selects all the attributes and display)

UPDATE BUS SET Bus No = 1 WHERE BUS NO=2;

Experiment 7: Querying

In this week you are going to practice queries (along with sub queries) using ANY, ALL, IN, Exists, NOT EXISTS, UNION, INTERSECT, Constraints etc.

Practice the following Queries:

1. Display unique PNR_no of all passengers.
2. Display all the names of male passengers.
3. Display the ticket numbers and names of all the passengers.
4. Find the ticket numbers of the passengers whose name start with 'r' and ends with 'h'.
5. Find the names of passengers whose age is between 30 and 45.
6. Display all the passengers names beginning with 'A'
7. Display the sorted list of passengers names

Experiment 8 and Experiment 9: Querying (continued...)

You are going to practice queries using Aggregate functions (COUNT, SUM, AVG, and MAX and MIN), GROUP BY, HAVING and Creation and dropping of Views.

1. Write a Query to display the Information present in the Passenger and cancellation tables. **Hint:** Use UNION Operator.
2. Display the number of days in a week on which the 9W01 bus is available.
3. Find number of tickets booked for each PNR_no using GROUP BY CLAUSE. **Hint:** Use GROUP BY on PNR_No.
4. Find the distinct PNR numbers that are present.
5. Find the number of tickets booked by a passenger where the number of seats is greater than 1. **Hint:** Use GROUP BY, WHERE and HAVING CLAUSES.
6. Find the total number of cancelled seats.

Experiment 10: Triggers

In this week you are going to work on Triggers. Creation of insert trigger, delete trigger, update trigger. Practice triggers using the above database.

Eg: **CREATE TRIGGER updcheck BEFORE UPDATE ON passenger**

```
FOR EACH ROW  
BEGIN  
IF NEW.TicketNO > 60 THEN  
SET New.Ticket no = Ticket no;  
ELSE  
SET New.Ticketno = 0;  
END IF;  
END;
```

Experiment 11: Procedures

In this session you are going to learn Creation of stored procedure, Execution of procedure and modification of procedure. Practice procedures using the above database.

Eg: **CREATE PROCEDURE myProc()**

```
BEGIN  
SELECT COUNT (Tickets) FROM Ticket WHERE age>=40;  
End;
```

Experiment 12: Cursors

In this week you need to do the following: Declare a cursor that defines a result set. Open the cursor to establish the result set. Fetch the data into local variables as needed from the cursor, one row at a time. Close the cursor when done

```
CREATE PROCEDURE myProc(in_customer_id INT)  
BEGIN  
DECLARE v_id INT;  
DECLARE v_name VARCHAR (30);  
DECLARE c1 CURSOR FOR SELECT stdId,stdFirstname FROM students WHERE  
stdId=in_customer_id;  
OPEN c1;  
FETCH c1 into v_id, v_name;  
Close c1;  
END;
```

Tables

BUS

Bus No: Varchar: PK (public key)

Source : Varchar

Destination : Varchar

Passenger

PPNO: Varchar (15)) : PK

Name: Varchar (15)

Age : int (4)

Sex:Char (10) : Male / Female

Address: VarChar (20)

Passenger_Tickets

PPNO: Varchar (15)) : PK

Ticket_No: Numeric (9)

Reservation

PNR_No: Numeric (9) : FK

Journey_date : datetime (8)

No_of_seats : int (8)

Address: Varchar (50)

Contact_No: Numeric (9) --> Should not be less than 9 and Should not accept any other character other than Integer

Status: Char (2) : Yes / No

Cancellation

PNR_No: Numeric(9) : FK

Journey_date : datetime(8)

No_of_seats : int (8)

Address : Varchar (50)

Contact_No: Numeric (9) --> Should not be less than 9 and Should not accept any other character other than Integer

Status: Char (2) : Yes / No

Ticket

Ticket_No: Numeric (9): PK

Journey_date : datetime(8)

Age : int (4)

Sex:Char(10) : Male / Female

Source : Varchar

Destination : Varchar

Dep_time : Varchar

REFERENCE BOOKS:

1. Introduction to SQL, Rick F. Vander Lans, Pearson education.
2. Oracle PL/SQL, B. Rosenzweig and E. Silvestrova, Pearson education.
3. SQL & PL/SQL for Oracle 10 g, Black Book, Dr. P. S. Deshpande, Dream Tech.
4. Oracle Database 11 g PL/SQL Programming, M. Mc Laughlin, TMH.

CS408ES: OPERATING SYSTEMS LAB

B.Tech. II Year II Sem.

L T P C
0 0 3 2

Course Objectives:

- To write programs in Linux environment using system calls.
- To implement the scheduling algorithms.
- To implement page replacement algorithms
- To implement file allocation methods.
- To understand and implement ipc mechanism using named and unnamed pipes.
- To develop solutions for synchronization problems using semaphores.

Course Outcomes:

- Ability to develop application programs using system calls in Unix.
- Ability to implement interprocess communication between two processes.
- Ability to design and solve synchronization problems.
- Ability to simulate and implement operating system concepts such as scheduling, deadlock management, file management, and memory management.

Use Linux operating system and GNU C compiler.

List of Programs:

1. Write C programs to simulate the following CPU scheduling algorithms:
a) Round Robin b) SJF
2. Write C programs to simulate the following CPU scheduling algorithms:
a) FCFS b) Priority
3. Write C programs to simulate the following File organization techniques:
a) Single level directory b) Two level c) Hierarchical
4. Write C programs to simulate the following File allocation methods:
a) Contiguous b) Linked c) Indexed
5. Write a C program to copy the contents of one file to another using system calls.
6. Write a C program to simulate Bankers Algorithm for Dead Lock Avoidance
7. Write a C program to simulate Bankers Algorithm for Dead Lock Prevention
8. Write C programs to simulate the following page replacement algorithms:
a) FIFO b) LRU c) LFU
9. Write C programs to simulate the following techniques of memory management:
a) Paging b) Segmentation
10. Write a C program to implement the ls | sort command. (Use unnamed Pipe)
11. Write a C program to solve the Dining- Philosopher problem using semaphores.
12. Write C programs to implement ipc between two unrelated processes using named pipe.

REFERENCE BOOKS:

1. An Introduction to Operating Systems, P.C.P Bhatt, 2nd edition, PHI.
2. Unix System Programming Using C++, Terrence Chan, PHI/Pearson.
3. Modern Operating Systems, Andrew S Tanenbaum, 3rd Edition, PHI

MC400HS: GENDER SENSITIZATION LAB

B.Tech. II Year II Sem.

L T P C
0 0 3 0

Course Objectives:

- To develop students' sensibility with regard to issues of gender in contemporary India.
- To provide a critical perspective on the socialization of men and women.
- To introduce students to information about some key biological aspects of genders.
- To expose the students to debates on the politics and economics of work.
- To help students reflect critically on gender violence.
- To expose students to more egalitarian interactions between men and women.

Course Outcomes:

- Students will have developed a better understanding of important issues related to gender in contemporary India.
- Students will be sensitized to basic dimensions of the biological, sociological, psychological and legal aspects of gender. This will be achieved through discussion of materials derived from research, facts, everyday life, literature, and film.
- Students will attain a finer grasp of how gender discrimination works in our society and how to counter it.
- Students will acquire insight into the gendered division of labour and its relation to politics and economics.
- Men and women students and professionals will be better equipped to work and live together as equals.
- Students will develop a sense of appreciation of women in all walks of life.
- Through providing accounts of studies and movements as well as the new laws that provide protection and relief to women, the textbook will empower students to understand and respond to gender violence.

UNIT - I

UNDERSTANDING GENDER

Gender: Why Should We Study It? (*Towards a World of Equals*: Unit -1)

Socialization: Making Women, Making Men (*Towards a World of Equals*: Unit -2)

Introduction. Preparing for Womanhood. Growing up Male. First lessons in Caste. Different Masculinities.

UNIT - II

GENDER AND BIOLOGY:

Missing Women: Sex Selection and Its Consequences (*Towards a World of Equals*: Unit -4)

Declining Sex Ratio. Demographic Consequences.

Gender Spectrum: Beyond the Binary (*Towards a World of Equals*: Unit -10)

Two or Many? Struggles with Discrimination.

UNIT - III

GENDER AND LABOUR

Housework: the Invisible Labour (*Towards a World of Equals*: Unit -3)

“My Mother doesn’t Work.” “Share the Load.”

Women’s Work: Its Politics and Economics (*Towards a World of Equals*: Unit -7)

Fact and Fiction. Unrecognized and Unaccounted work. Additional Reading: Wages and Conditions of Work.

UNIT-IV

ISSUES OF VIOLENCE

Sexual Harassment: Say No! (*Towards a World of Equals*: Unit -6)

Sexual Harassment, not Eve-teasing- Coping with Everyday Harassment- Further Reading: “Chupulu”.

Domestic Violence: Speaking Out (*Towards a World of Equals*: Unit -8)

Is Home a Safe Place? -When Women Unite [Film]. Rebuilding Lives. Additional Reading: New Forums for Justice.

Thinking about Sexual Violence (*Towards a World of Equals*: Unit -11)

Blaming the Victim-“I Fought for my Life....” - Additional Reading: The Caste Face of Violence.

UNIT - V

GENDER: CO - EXISTENCE

Just Relationships: Being Together as Equals (*Towards a World of Equals*: Unit -12)

Mary Kom and Onler. Love and Acid just do not Mix. Love Letters. Mothers and Fathers. Additional Reading: Rosa Parks-The Brave Heart.

TEXTBOOK

All the five Units in the Textbook, “*Towards a World of Equals: A Bilingual Textbook on Gender*” written by A. Suneetha, Uma Bhrugubanda, Duggirala Vasanta, Rama Melkote, Vasudha Nagaraj, Asma Rasheed, Gogu Shyamala, Deepa Sreenivas and Susie Tharu and published by **Telugu Akademi, Hyderabad**, Telangana State in the year **2015**.

Note: Since it is an Interdisciplinary Course, Resource Persons can be drawn from the fields of English Literature or Sociology or Political Science or any other qualified faculty who has expertise in this field from engineering departments.

REFERENCE BOOKS:

1. Menon, Nivedita. Seeing like a Feminist. New Delhi: Zubaan-Penguin Books, 2012
2. Abdulali Sohaila. “*I Fought For My Life...and Won.*” Available online at:
<http://www.thealternative.in/lifestyle/i-fought-for-my-lifeand-won-sohaila-abdulal/>

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD**B.TECH. COMPUTER SCIENCE AND ENGINEERING
III YEAR COURSE STRUCTURE & SYLLABUS (R16)****Applicable From 2016-17 Admitted Batch****III YEAR I SEMESTER**

S. No	Course Code	Course Title	L	T	P	Credits
1	CS501PC	Design and Analysis of Algorithms	4	0	0	4
2	CS502PC	Data Communication and Computer Networks	4	0	0	4
3	CS503PC	Software Engineering	4	0	0	4
4	SM504MS	Fundamentals of Management	3	0	0	3
5		Open Elective –I	3	0	0	3
6	CS505PC	Design and Analysis of Algorithms Lab	0	0	3	2
7	CS506PC	Computer Networks Lab	0	0	3	2
8	CS507PC	Software Engineering Lab	0	0	3	2
9	*MC500HS	Professional Ethics	3	0	0	0
		Total Credits	21	0	9	24

III YEAR II SEMESTER

S. No	Course Code	Course Title	L	T	P	Credits
1	CS601PC	Compiler Design	4	0	0	4
2	CS602PC	Web Technologies	4	0	0	4
3	CS603PC	Cryptography and Network Security	4	0	0	4
4		Open Elective-II	3	0	0	3
5		Professional Elective-I	3	0	0	3
6	CS604PC	Cryptography and Network Security Lab	0	0	3	2
7	CS605PC	Web Technologies Lab	0	0	3	2
8	EN606HS	Advanced English Communication Skills Lab	0	0	3	2
		Total Credits	18	0	9	24

During Summer Vacation between III and IV Years: Industry Oriented Mini Project

Professional Elective – I

CS611PE	Mobile Computing
CS612PE	Design Patterns
CS613PE	Artificial Intelligence
CS614PE	Information Security Management (Security Analyst - I)
CS615PE	Introduction to Analytics (Associate Analytics - I)

***Open Elective** subjects' syllabus is provided in a separate document.

***Open Elective** – Students should take Open Electives from the List of Open Electives Offered by Other Departments/Branches Only.

Ex: - A Student of Mechanical Engineering can take Open Electives from all other departments/branches except Open Electives offered by Mechanical Engineering Dept.

DESIGN AND ANALYSIS OF ALGORITHMS

B.Tech. III Year I Sem.
Course Code: CS501PC

L	T	P	C
4	0	0	4

Course Objectives

- To analyze performance of algorithms.
- To choose the appropriate data structure and algorithm design method for a specified application.
- To understand how the choice of data structures and algorithm design methods impacts the performance of programs.
- To solve problems using algorithm design methods such as the greedy method, divide and conquer, dynamic programming, backtracking and branch and bound.
- To understand the differences between tractable and intractable problems.
- To introduce P and NP classes.

Course Outcomes:

- Ability to analyze the performance of algorithms.
- Ability to choose appropriate algorithm design techniques for solving problems.
- Ability to understand how the choice of data structures and the algorithm design methods impact the performance of programs.

UNIT - I

Introduction-Algorithm definition, Algorithm Specification, Performance Analysis-Space complexity, Time complexity, Randomized Algorithms.

Divide and conquer- General method, applications - Binary search, Merge sort, Quick sort, Strassen's Matrix Multiplication.

UNIT - II

Disjoint set operations, union and find algorithms, AND/OR graphs, Connected Components and Spanning trees, Bi-connected components **Backtracking**-General method, applications-The 8-queen problem, sum of subsets problem, graph coloring, Hamiltonian cycles.

UNIT - III

Greedy method- General method, applications- Knapsack problem, Job sequencing with deadlines, Minimum cost spanning trees, Single source shortest path problem.

UNIT - IV

Dynamic Programming- General Method, applications- Chained matrix multiplication, All pairs shortest path problem, Optimal binary search trees, 0/1 knapsack problem, Reliability design, Traveling sales person problem.

UNIT - V

Branch and Bound- General Method, applications-0/1 Knapsack problem, LC Branch and Bound solution, FIFO Branch and Bound solution, Traveling sales person problem.

NP-Hard and NP-Complete problems- Basic concepts, Non-deterministic algorithms, NP - Hard and NP- Complete classes, Cook's theorem.

TEXT BOOKS:

1. Fundamentals of Computer Algorithms, 2nd Edition, Ellis Horowitz, Sartaj Sahni and S. Rajasekharan, Universities Press.
2. Design and Analysis of Algorithms, P. H. Dave, H.B.Dave, 2nd edition, Pearson Education.

REFERENCE BOOKS

1. Algorithm Design: Foundations, Analysis and Internet examples, M. T. Goodrich and R. Tomassia, John Wiley and sons.
2. Design and Analysis of Algorithms, S. Sridhar, Oxford Univ. Press
3. Design and Analysis of algorithms, Aho, Ullman and Hopcroft, Pearson Education.
4. Foundations of Algorithms,, R. Neapolitan and K. Naimipour, 4th edition, Jones and Bartlett Student edition.
5. Introduction to Algorithms, 3rd Edition, T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, PHI

DATA COMMUNICATION AND COMPUTER NETWORKS

B.Tech. III Year I Sem.
Course Code: CS502PC

L	T	P	C
4	0	0	4

Course Objectives:

- To introduce the fundamental various types of computer networks.
- To demonstrate the TCP/IP and OSI models with merits and demerits.
- To explore the various layers of OSI Model.
- To introduce UDP and TCP Models.

Course Outcomes:

- Students should be understand and explore the basics of Computer Networks and Various Protocols. He/She will be in a position to understand the World Wide Web concepts.
- Students will be in a position to administrate a network and flow of information further he/she can understand easily the concepts of network security, Mobile and ad hoc networks.

UNIT – I

Data Communications: Components – Direction of Data flow – Networks – Components and Categories – Types of Connections – Topologies –Protocols and Standards – ISO / OSI model, Example Networks such as ATM, Frame Relay, ISDN Physical layer: Transmission modes, Multiplexing, Transmission Media, Switching, Circuit Switched Networks, Datagram Networks, Virtual Circuit Networks.

UNIT – II

Data link layer: Introduction, Framing, and Error – Detection and Correction – Parity – LRC – CRC Hamming code, Flow and Error Control, Noiseless Channels, Noisy Channels, HDLC, Point to Point Protocols. 111 Medium Access sub layer: ALOHA, CSMA/CD, LAN – Ethernet IEEE 802.3, IEEE 802.5 – IEEE 802.11, Random access, Controlled access, Channelization.

UNIT – III

Network layer: Logical Addressing, Internetworking, Tunneling, Address mapping, ICMP, IGMP, Forwarding, Uni-Cast Routing Protocols, Multicast Routing Protocols.

UNIT – IV

Transport Layer: Process to Process Delivery, UDP and TCP protocols, Data Traffic, Congestion, Congestion Control, QoS, Integrated Services, Differentiated Services, QoS in Switched Networks.

UNIT – V

Application Layer: Domain name space, DNS in internet, electronic mail, SMTP, FTP, WWW, HTTP, SNMP.

TEXT BOOKS:

1. Data Communications and Networking, Behrouz A. Forouzan , Fourth Edition TMH, 2006.
2. Computer Networks, Andrew S Tanenbaum, 4th Edition. Pearson Education, PHI.

REFERENCES:

1. Data communications and Computer Networks, P.C .Gupta, PHI.
2. An Engineering Approach to Computer Networks, S. Keshav, 2nd Edition, Pearson Education.
3. Understanding communications and Networks, 3rd Edition, W.A. Shay, Cengage Learning.
4. Computer Networking: A Top-Down Approach Featuring the Internet. James F. Kurose & Keith W. Ross, 3rd Edition, Pearson Education.
5. Data and Computer Communication, William Stallings, Sixth Edition, Pearson Education, 2000

SOFTWARE ENGINEERING

B.Tech. III Year I Sem.
Course Code: CS503PC

L	T	P	C
4	0	0	4

Course Objectives:

- To understanding of software process models such as waterfall and evolutionary models.
- To understanding of software requirements and SRS document.
- To understanding of different software architectural styles.
- To understanding of software testing approaches such as unit testing and integration testing.
- To understanding on quality control and how to ensure good quality software.

Course Outcomes:

- Ability to identify the minimum requirements for the development of application.
- Ability to develop, maintain, efficient, reliable and cost effective software solutions
- Ability to critically thinking and evaluate assumptions and arguments.

UNIT- I

Introduction to Software Engineering: The evolving role of software, Changing Nature of Software, legacy software, Software myths.

A Generic view of process: Software engineering- A layered technology, a process framework, The Capability Maturity Model Integration (CMMI), Process patterns, process assessment, personal and team process models.

Process models: The waterfall model, Incremental process models, Evolutionary process models, Specialized process models, The Unified process.

UNIT- II

Software Requirements: Functional and non-functional requirements, User requirements, System requirements, Interface specification, the software requirements document.

Requirements engineering process: Feasibility studies, Requirements elicitation and analysis, Requirements validation, Requirements management.

System models: Context Models, Behavioral models, Data models, Object models, structured methods.

UNIT- III

Design Engineering: Design process and Design quality, Design concepts, the design model, pattern based software design.

Creating an architectural design: software architecture, Data design, Architectural styles and patterns, Architectural Design, assessing alternative architectural designs, mapping data flow into a software architecture.

Modeling component-level design: Designing class-based components, conducting component-level design, object constraint language, designing conventional components.
Performing User interface design: Golden rules, User interface analysis, and design, interface analysis, interface design steps, Design evaluation.

UNIT- IV

Testing Strategies: A strategic approach to software testing, test strategies for conventional software, Black-Box and White-Box testing, Validation testing, System testing, the art of Debugging.

Product metrics: Software Quality, Frame work for Product metrics, Metrics for Analysis Model, Metrics for Design Model, Metrics for source code, Metrics for testing, Metrics for maintenance.

Metrics for Process and Products: Software Measurement, Metrics for software quality.

UNIT- V

Risk management: Reactive vs Proactive Risk strategies, software risks, Risk identification, Risk projection, Risk refinement, RMMM, RMMM Plan.

Quality Management: Quality concepts, Software quality assurance, Software Reviews, Formal technical reviews, Statistical Software quality Assurance, Software reliability, The ISO 9000 quality standards.

TEXT BOOKS:

1. Software engineering A practitioner's Approach, Roger S Pressman, sixth edition McGraw Hill International Edition.
2. Software Engineering, Ian Sommerville, seventh edition, Pearson education.

REFERENCE BOOKS:

1. Software Engineering, A Precise Approach, Pankaj Jalote, Wiley India, 2010.
2. Software Engineering : A Primer, Waman S Jawadekar, Tata McGraw-Hill, 2008
3. Fundamentals of Software Engineering, Rajib Mall, PHI, 2005
4. Software Engineering, Principles and Practices, Deepak Jain, Oxford University Press.
5. Software Engineering1: Abstraction and modeling, Diner Bjerne, Springer International edition, 2006.
6. Software Engineering2: Specification of systems and languages, Diner Bjerne, Springer International edition 2006.
7. Software Engineering Foundations, Yingxu Wang, Auerbach Publications, 2008.
8. Software Engineering Principles and Practice, Hans Van Vliet, 3rd edition, John Wiley & Sons Ltd.
9. Software Engineering 3: Domains, Requirements, and Software Design, D. Bjerne, Springer International Edition.
10. Introduction to Software Engineering, R. J. Leach, CRC Press.

FUNDAMENTALS OF MANAGEMENT

B.Tech. III Year I Sem.
Course Code: SM504MS

L	T	P	C
3	0	0	3

Course Objective: To understand the Management Concepts, applications of Concepts in Practical aspects of business and development of Managerial Skills.

Course Outcome: The students understand the significance of Management in their Profession. The various Management Functions like Planning, Organizing, Staffing, Leading, Motivation and Control aspects are learnt in this course. The students can explore the Management Practices in their domain area.

UNIT - I

Introduction to Management: Definition, Nature and Scope, Functions, Managerial Roles, Levels of Management, Managerial Skills, Challenges of Management; Evolution of Management- Classical Approach- Scientific and Administrative Management; The Behavioral approach; The Quantitative approach; The Systems Approach; Contingency Approach, IT Approach.

UNIT - II

Planning and Decision Making: General Framework for Planning - Planning Process, Types of Plans, Management by Objectives; Development of Business Strategy. Decision making and Problem Solving - Programmed and Non Programmed Decisions, Steps in Problem Solving and Decision Making; Bounded Rationality and Influences on Decision Making; Group Problem Solving and Decision Making, Creativity and Innovation in Managerial Work.

UNIT - III

Organization and HRM: Principles of Organization: Organizational Design & Organizational Structures; Departmentalization, Delegation; Empowerment, Centralization, Decentralization, Recentralization; Organizational Culture; Organizational Climate and Organizational Change.

Human Resource Management & Business Strategy: Talent Management, Talent Management Models and Strategic Human Resource Planning; Recruitment and Selection; Training and Development; Performance Appraisal.

UNIT - IV

Leading and Motivation: Leadership, Power and Authority, Leadership Styles; Behavioral Leadership, Situational Leadership, Leadership Skills, Leader as Mentor and Coach, Leadership during adversity and Crisis; Handling Employee and Customer Complaints, Team Leadership.

Motivation - Types of Motivation; Relationship between Motivation, Performance and Engagement, Content Motivational Theories - Needs Hierarchy Theory, Two Factor Theory, Theory X and Theory Y.

UNIT - V

Controlling: Control, Types and Strategies for Control, Steps in Control Process, Budgetary and Non- Budgetary Controls. Characteristics of Effective Controls, Establishing control systems, Control frequency and Methods.

Text Books:

1. Management Fundamentals, Robert N Lussier, 5e, Cengage Learning, 2013.
2. Fundamentals of Management, Stephen P. Robbins, Pearson Education, 2009.

References:

1. Essentials of Management, Koontz Kleihrich, Tata McGraw Hill.
2. Management Essentials, Andrew DuBrin, 9e, Cengage Learning, 2012.

DESIGN AND ANALYSIS OF ALGORITHMS LAB

B.Tech. III Year I Sem.
Course Code: CS505PC

L	T	P	C
0	0	3	2

Course Objectives:

- To write programs in java to solve problems using divide and conquer strategy.
- To write programs in java to solve problems using backtracking strategy.
- To write programs in java to solve problems using greedy and dynamic programming techniques.

Course Outcomes:

- Ability to write programs in java to solve problems using algorithm design techniques such as Divide and Conquer, Greedy, Dynamic programming, and Backtracking.

List of Experiments:

1. Write a java program to implement Quick sort algorithm for sorting a list of integers in ascending order
2. Write a java program to implement Merge sort algorithm for sorting a list of integers in ascending order.
3. i) Write a java program to implement the dfs algorithm for a graph.
4. ii) Write a java program to implement the bfs algorithm for a graph.
5. Write a java programs to implement backtracking algorithm for the N-queens problem.
6. Write a java program to implement the backtracking algorithm for the sum of subsets problem.
7. Write a java program to implement the backtracking algorithm for the Hamiltonian Circuits problem.
8. Write a java program to implement greedy algorithm for job sequencing with deadlines.
9. Write a java program to implement Dijkstra's algorithm for the Single source shortest path problem.
10. Write a java program that implements Prim's algorithm to generate minimum cost spanning tree.
11. Write a java program that implements Kruskal's algorithm to generate minimum cost spanning tree
12. Write a java program to implement Floyd's algorithm for the all pairs shortest path problem.
13. Write a java program to implement Dynamic Programming algorithm for the 0/1 Knapsack problem.
14. Write a java program to implement Dynamic Programming algorithm for the Optimal Binary Search Tree Problem.

REFERENCE BOOKS

1. Data structures, Algorithms and Applications in java, 2nd Edition, S. Sahani, Universities Press.
2. Data structures and Algorithms in java, 3rd edition, A. Drozdek, Cengage Learning.
3. Data structures with Java, J. R. Hubbard, 2nd edition, Schaum's Outlines, TMH.
4. Data structures and algorithms in Java, 2nd Edition, R. Lafore, Pearson Education.
5. Data Structures using Java, D. S. Malik and P.S. Nair, Cengage Learning.

COMPUTER NETWORKS LAB

B.Tech. III Year I Sem.
Course Code: CS506PC

L	T	P	C
0	0	3	2

Course Objectives:

- To Understand the functionalities of various layers of OSI model
- To understand the operating System functionalities

Course Outcomes:

- Ability to understand the encryption and decryption concepts in Linux environment
- Ability to apply appropriate algorithm for the finding of shortest route.
- Ability to configure the routing table

System/ Software Requirement

- Intel based desktop PCs LAN CONNECTED with minimum of 166 MHZ or faster processor with at least 64 MB RAM and 100 MB free disk space

Computer Networks Lab:

1. Implement the data link layer framing methods such as character, character stuffing, and bit stuffing.
2. Implement on a data set of characters the three CRC polynomials – CRC 12, CRC 16 and CRC CCIP .
3. Implement Dijkstra's algorithm to compute the Shortest path thru a graph.
4. Take an example subnet graph with weights indicating delay between nodes. Now obtain Routing table at each node using distance vector routing algorithm
5. Take an example subnet of hosts. Obtain broadcast tree for it.
6. Take a 64 bit playing text and encrypt the same using DES algorithm.
7. Write a program to break the above DES coding
8. Using RSA algorithm encrypts a text data and Decrypt the same.

SOFTWARE ENGINEERING LAB

B.Tech. III Year I Sem.
Course Code: CS507PC

L	T	P	C
0	0	3	2

Course Objectives:

- To understand the software engineering methodologies involved in the phases for project development.
- To gain knowledge about open source tools used for implementing software engineering methods.
- To exercise developing product-startups implementing software engineering methods.
- Open source Tools: StarUML / UMLGraph / Topcased

Prepare the following documents and develop the software project startup, prototype model, using software engineering methodology for at least two real time scenarios or for the sample experiments.

- Problem Analysis and Project Planning -Thorough study of the problem – Identify Project scope, Objectives and Infrastructure.
- Software Requirement Analysis – Describe the individual Phases/modules of the project and Identify deliverables. Identify functional and non-functional requirements.
- Data Modeling – Use work products – data dictionary.
- Software Designing - Develop use case diagrams and activity diagrams, build and test class diagrams, sequence diagrams and add interface to class diagrams.
- Prototype model – Develop the prototype of the product.

The SRS and prototype model should be submitted for end semester examination.

List of Sample Experiments:

1. Course management system (CMS)

A course management system (CMS) is a collection of software tools providing an online environment for course interactions. A CMS typically includes a variety of online tools and environments, such as:

- An area for faculty posting of class materials such as course syllabus and handouts
- An area for student posting of papers and other assignments
- A grade book where faculty can record grades and each student can view his or her grades
- An integrated email tool allowing participants to send announcement email messages to the entire class or to a subset of the entire class
- A chat tool allowing synchronous communication among class participants

- A threaded discussion board allowing asynchronous communication among participants

In addition, a CMS is typically integrated with other databases in the university so that students enrolled in a particular course are automatically registered in the CMS as participants in that course.

The Course Management System (CMS) is a web application for department personnel, Academic Senate, and Registrar staff to view, enter, and manage course information formerly submitted via paper.

Departments can use CMS to create new course proposals, submit changes for existing courses, and track the progress of proposals as they move through the stages of online approval.

2. Easy Leave

This project is aimed at developing a web based Leave Management Tool, which is of importance to either an organization or a college.

The **Easy Leave** is an Intranet based application that can be accessed throughout the organization or a specified group/Dept. This system can be used to automate the workflow of leave applications and their approvals. The periodic crediting of leave is also automated. There are features like notifications, cancellation of leave, automatic approval of leave, report generators etc in this Tool.

Functional components of the project:

There are registered people in the system. Some are approvers. An approver can also be a requestor. In an organization, the hierarchy could be Engineers/Managers/Business Managers/Managing Director etc. In a college, it could be Lecturer/Professor/Head of the Department/Dean/Principal etc.

Following is a list of functionalities of the system: A person should be able to

- login to the system through the first page of the application
- change the password after logging into the system
- see his/her eligibility details (like how many days of leave he/she is eligible for etc)
- query the leave balance
- see his/her leave history since the time he/she joined the company/college
- apply for leave, specifying the from and to dates, reason for taking leave, address for communication while on leave and his/her superior's email id
- see his/her current leave applications and the leave applications that are submitted to him/her for approval or cancellation
- approve/reject the leave applications that are submitted to him/her
- withdraw his/her leave application (which has not been approved yet)
- Cancel his/her leave (which has been already approved). This will need to be approved by his/her Superior

- get help about the leave system on how to use the different features of the system
- As soon as a leave application /cancellation request /withdrawal /approval /rejection /password-change is made by the person, an automatic email should be sent to the person and his superior giving details about the action
- The number of days of leave (as per the assumed leave policy) should be automatically credited to everybody and a notification regarding the same be sent to them automatically
- An automatic leave-approval facility for leave applications which are older than 2 weeks should be there. Notification about the automatic leave approval should be sent to the person as well as his superior

3. E-Bidding

Auctions are among the latest economic institutions in place. They have been used since antiquity to sell a wide variety of goods, and their basic form has remained unchanged. In this dissertation, we explore the efficiency of common auctions when values are interdependent-the value to a particular bidder may depend on information available only to others-and asymmetric. In this setting, it is well known that sealed-bid auctions do not achieve efficient allocations in general since they do not allow the information held by different bidders to be shared.

Typically, in an auction, say of the kind used to sell art, the auctioneer sets a relatively low initial price. This price is then increased until only one bidder is willing to buy the object, and the exact manner in which this is done varies. In my model a bidder who drops out at some price can "reenter" at a higher price.

With the invention of E-commerce technologies over the Internet the opportunity to bid from the comfort of one's own home has seen a change like never seen before. Within the span of a few short years, what may have began as an experimental idea has grown to an immensely popular hobby, and in some cases, a means of livelihood, the Auction Patrol gathers tremendous response every day, all day. With the point and click of the mouse, one may bid on an item they may need or just want, and in moments they find that either they are the top bidder or someone else wants it more, and you're outbid! The excitement of an auction all from the comfort of home is a completely different experience.

Society cannot seem to escape the criminal element in the physical world, and so it is the same with Auction Patrols. This is one area where in a question can be raised as to how safe Auction Patrols.

Proposed system

To generate the quick reports

To make accuracy and efficient calculations

To provide proper information briefly

To provide data security

To provide huge maintenance of records
Flexibility of transactions can be completed in time

4. Electronic Cash counter

This project is mainly developed for the Account Division of a Banking sector to provide better interface of the entire banking transactions. This system is aimed to give a better outlook to the user interfaces and to implement all the banking transactions like:

- Supply of Account Information
- New Account Creations
- Deposits
- Withdraws
- Cheque book issues
- Stop payments
- Transfer of accounts
- Report Generations.

Proposed System:

The development of the new system contains the following activities, which try to automate the entire process keeping in view of the database integration approach.

- User friendliness is provided in the application with various controls.
- The system makes the overall project management much easier and flexible.
- Readily upload the latest updates, allows user to download the alerts by clicking the URL.
- There is no risk of data mismanagement at any level while the project development is under process.
- It provides high level of security with different level of authentication

PROFESSIONAL ETHICS

B.Tech. III Year I Sem.
Course Code: MC500HS

L	T	P	C
3	0	0	0

Course Objective: To enable the students to imbibe and internalize the Values and Ethical Behaviour in the personal and Professional lives.

Course Outcome: The students will understand the importance of Values and Ethics in their personal lives and professional careers. The students will learn the rights and responsibilities as an employee, team member and a global citizen.

UNIT - I

Introduction to Professional Ethics: Basic Concepts, Governing Ethics, Personal & Professional Ethics, Ethical Dilemmas, Life Skills, Emotional Intelligence, Thoughts of Ethics, Value Education, Dimensions of Ethics, Profession and professionalism, Professional Associations, Professional Risks, Professional Accountabilities, Professional Success, Ethics and Profession.

UNIT - II

Basic Theories: Basic Ethical Principles, Moral Developments, Deontology, Utilitarianism, Virtue Theory, Rights Theory, Casuist Theory, Moral Absolution, Moral Rationalism, Moral Pluralism, Ethical Egoism, Feminist Consequentialism, Moral Issues, Moral Dilemmas, Moral Autonomy.

UNIT - III

Professional Practices in Engineering: Professions and Norms of Professional Conduct, Norms of Professional Conduct vs. Profession; Responsibilities, Obligations and Moral Values in Professional Ethics, Professional codes of ethics, the limits of predictability and responsibilities of the engineering profession.

Central Responsibilities of Engineers - The Centrality of Responsibilities of Professional Ethics; lessons from 1979 American Airlines DC-10 Crash and Kansas City Hyatt Regency Walk away Collapse.

UNIT - IV

Work Place Rights & Responsibilities, Ethics in changing domains of Research, Engineers and Managers; Organizational Complaint Procedure, difference of Professional Judgment within the Nuclear Regulatory Commission (NRC), the Hanford Nuclear Reservation.

Ethics in changing domains of research - The US government wide definition of research misconduct, research misconduct distinguished from mistakes and errors, recent history of attention to research misconduct, the emerging emphasis on understanding and fostering responsible conduct, responsible authorship, reviewing & editing.

UNIT - V

Global issues in Professional Ethics: Introduction – Current Scenario, Technology Globalization of MNCs, International Trade, World Summits, Issues, Business Ethics and Corporate Governance, Sustainable Development Ecosystem, Energy Concerns, Ozone Deflection, Pollution, Ethics in Manufacturing and Marketing, Media Ethics; War Ethics; Bio Ethics, Intellectual Property Rights.

TEXT BOOKS:

1. Professional Ethics: R. Subramanian, Oxford University Press, 2015.
2. Ethics in Engineering Practice & Research, Caroline Whitbeck, 2e, Cambridge University Press 2015.

REFERENCES:

1. Engineering Ethics, Concepts Cases: Charles E Harris Jr., Michael S Pritchard, Michael J Rabins, 4e , Cengage learning, 2015.
2. Business Ethics concepts & Cases: Manuel G Velasquez, 6e, PHI, 2008.

COMPILER DESIGN

B.Tech. III Year II Sem.
Course Code: CS601PC

L	T	P	C
4	0	0	4

Course Objectives:

- To understand the various phases in the design of a compiler.
- To understand the design of top-down and bottom-up parsers.
- To understand syntax directed translation schemes.
- To introduce lex and yacc tools.
- To learn to develop algorithms to generate code for a target machine.

Course Outcomes:

- Ability to design, develop, and implement a compiler for any language.
- Able to use lex and yacc tools for developing a scanner and a parser.
- Able to design and implement LL and LR parsers.
- Able to design algorithms to perform code optimization in order to improve the performance of a program in terms of space and time complexity.
- Ability to design algorithms to generate machine code

UNIT - I

Introduction: Language Processors, the structure of a compiler, the science of building a compiler, programming language basics.

Lexical Analysis: The Role of the Lexical Analyzer, Input Buffering, Recognition of Tokens, The Lexical-Analyzer Generator Lex, Finite Automata, From Regular Expressions to Automata, Design of a Lexical-Analyzer Generator, Optimization of DFA-Based Pattern Matchers.

UNIT - II

Syntax Analysis: Introduction, Context-Free Grammars, Writing a Grammar, Top-Down Parsing, Bottom-Up Parsing, Introduction to LR Parsing: Simple LR, More Powerful LR Parsers, Using Ambiguous Grammars, Parser Generators.

UNIT - III

Syntax-Directed Translation: Syntax-Directed Definitions, Evaluation Orders for SDD's, Applications of Syntax-Directed Translation, Syntax-Directed Translation Schemes, and Implementing L-Attributed SDD's.

Intermediate-Code Generation: Variants of Syntax Trees, Three-Address Code, Types and Declarations, Type Checking, Control Flow, Back patching, Switch-Statements, Intermediate Code for Procedures.

UNIT - IV

Run-Time Environments: Storage organization, Stack Allocation of Space, Access to Nonlocal Data on the Stack, Heap Management, Introduction to Garbage Collection, Introduction to Trace-Based Collection.

Code Generation: Issues in the Design of a Code Generator, The Target Language, Addresses in the Target Code, Basic Blocks and Flow Graphs, Optimization of Basic Blocks, A Simple Code Generator, Peephole Optimization, Register Allocation and Assignment, Dynamic Programming Code-Generation.

UNIT - V

Machine-Independent Optimizations: The Principal Sources of Optimization, Introduction to Data-Flow Analysis, Foundations of Data-Flow Analysis, Constant Propagation, Partial-Redundancy Elimination, Loops in Flow Graphs.

TEXT BOOKS

1. Compilers: Principles, Techniques and Tools, Second Edition, Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman, Pearson.

REFERENCE BOOKS

1. Compiler Construction-Principles and Practice, Kenneth C Louden, Cengage Learning.
2. Modern compiler implementation in C, Andrew W Appel, Revised edition, Cambridge University Press.
3. The Theory and Practice of Compiler writing, J. P. Tremblay and P. G. Sorenson, TMH
4. Writing compilers and interpreters, R. Mak, 3rd edition, Wiley student edition.
5. lex & yacc – John R. Levine, Tony Mason, Doug Brown, O'reilly

WEB TECHNOLOGIES

B.Tech. III Year II Sem.
Course Code: CS602PC

L	T	P	C
4	0	0	4

Course Objectives:

- To introduce PHP language for server side scripting
- To introduce XML and processing of XML Data with Java
- To introduce Server side programming with Java Servlets and JSP
- To introduce Client side scripting with Javascript and AJAX.

Course Outcomes:

- gain knowledge of client side scripting, validation of forms and AJAX programming
- have understanding of server side scripting with PHP language
- have understanding of what is XML and how to parse and use XML Data with Java
- To introduce Server side programming with Java Servlets and JSP

UNIT - I

Introduction to PHP: Declaring variables, data types, arrays, strings, operators, expressions, control structures, functions, Reading data from web form controls like text boxes, radio buttons, lists etc., Handling File Uploads, Connecting to database (MySQL as reference), executing simple queries, handling results, Handling sessions and cookies

File Handling in PHP: File operations like opening, closing, reading, writing, appending, deleting etc. on text and binary files, listing directories

UNIT - II

XML: Introduction to XML, Defining XML tags, their attributes and values, Document Type Definition, XML Schemas, Document Object Model, XHTML

Parsing XML Data - DOM and SAX Parsers in java.

UNIT - III

Introduction to Servlets: Common Gateway Interface (CGI), Lifecycle of a Servlet, deploying a servlet, The Servlet API, Reading Servlet parameters, Reading Initialization parameters, Handling Http Request & Responses, Using Cookies and Sessions, connecting to a database using JDBC.

UNIT - IV

Introduction to JSP: The Anatomy of a JSP Page, JSP Processing, Declarations, Directives, Expressions, Code Snippets, implicit objects, Using Beans in JSP Pages, Using Cookies and session for session tracking, connecting to database in JSP.

UNIT- V

Client side Scripting: Introduction to Javascript: Javascript language - declaring variables, scope of variables, functions, event handlers (onclick, onsubmit etc.), Document Object Model, Form validation. Simple AJAX application.

TEXT BOOKS:

1. Web Technologies, Uttam K Roy, Oxford University Press
2. The Complete Reference PHP – Steven Holzner, Tata McGraw-Hill

REFERENCE BOOKS:

1. Web Programming, building internet applications, Chris Bates 2nd edition, Wiley Dreamtech
2. Java Server Pages –Hans Bergsten, SPD O'Reilly
3. Java Script, D. Flanagan, O'Reilly,SPD.
4. Beginning Web Programming-Jon Duckett WROX.
5. Programming World Wide Web, R. W. Sebesta, Fourth Edition, Pearson.
6. Internet and World Wide Web – How to program, Dietel and Nieto, Pearson.

CRYPTOGRAPHY AND NETWORK SECURITY

B.Tech. III Year II Sem.
Course Code: CS603PC

L	T	P	C
4	0	0	4

Course Objectives:

- Explain the objectives of information security
- Explain the importance and application of each of confidentiality, integrity, authentication and availability
- Understand various cryptographic algorithms.
- Understand the basic categories of threats to computers and networks
- Describe public-key cryptosystem.
- Describe the enhancements made to IPv4 by IPSec
- Understand Intrusions and intrusion detection
- Discuss the fundamental ideas of public-key cryptography.
- Generate and distribute a PGP key pair and use the PGP package to send an encrypted e-mail message.
- Discuss Web security and Firewalls

Course Outcomes:

- Student will be able to understand basic cryptographic algorithms, message and web authentication and security issues.
- Ability to identify information system requirements for both of them such as client and server.
- Ability to understand the current legal issues towards information security.

UNIT – I

Security Concepts: Introduction, The need for security, Security approaches, Principles of security, Types of Security attacks, Security services, Security Mechanisms, A model for Network Security

Cryptography Concepts and Techniques: Introduction, plain text and cipher text, substitution techniques, transposition techniques, encryption and decryption, symmetric and asymmetric key cryptography, steganography, key range and key size, possible types of attacks.

UNIT – II

Symmetric key Ciphers: Block Cipher principles, DES, AES, Blowfish, RC5, IDEA, Block cipher operation, Stream ciphers, RC4.

Asymmetric key Ciphers: Principles of public key cryptosystems, RSA algorithm, Elgamal Cryptography, Diffie-Hellman Key Exchange, Knapsack Algorithm.

UNIT – III

Cryptographic Hash Functions: Message Authentication, Secure Hash Algorithm (SHA-512), **Message authentication codes:** Authentication requirements, HMAC, CMAC, Digital signatures, Elgamal Digital Signature Scheme.

Key Management and Distribution: Symmetric Key Distribution Using Symmetric & Asymmetric Encryption, Distribution of Public Keys, Kerberos, X.509 Authentication Service, Public – Key Infrastructure

UNIT – IV

Transport-level Security: Web security considerations, Secure Socket Layer and Transport Layer Security, HTTPS, Secure Shell (SSH)

Wireless Network Security: Wireless Security, Mobile Device Security, IEEE 802.11 Wireless LAN, IEEE 802.11i Wireless LAN Security

UNIT – V

E-Mail Security: Pretty Good Privacy, S/MIME **IP Security:** IP Security overview, IP Security architecture, Authentication Header, Encapsulating security payload, Combining security associations, Internet Key Exchange

Case Studies on Cryptography and security: Secure Multiparty Calculation, Virtual Elections, Single sign On, Secure Inter-branch Payment Transactions, Cross site Scripting Vulnerability.

TEXT BOOKS:

1. Cryptography and Network Security - Principles and Practice: William Stallings, Pearson Education, 6th Edition
2. Cryptography and Network Security: Atul Kahate, Mc Graw Hill, 3rd Edition

REFERENCE BOOKS:

1. Cryptography and Network Security: C K Shyamala, N Harini, Dr T R Padmanabhan, Wiley India, 1st Edition.
2. Cryptography and Network Security : Forouzan Mukhopadhyay, Mc Graw Hill, 3rd Edition
3. Information Security, Principles, and Practice: Mark Stamp, Wiley India.
4. Principles of Computer Security: WM. Arthur Conklin, Greg White, TMH
5. Introduction to Network Security: Neal Krawetz, CENGAGE Learning
6. Network Security and Cryptography: Bernard Menezes, CENGAGE Learning

**MOBILE COMPUTING
(PROFESSIONAL ELECTIVE - I)**

**B.Tech. III Year II Sem.
Course Code: CS611PE**

L	T	P	C
3	0	0	3

Course Objectives:

- To make the student understand the concept of mobile computing paradigm, its novel applications and limitations.
- To understand the typical mobile networking infrastructure through a popular GSM protocol
- To understand the issues and solutions of various layers of mobile networks, namely MAC layer, Network Layer & Transport Layer
- To understand the database issues in mobile environments & data delivery models.
- To understand the ad hoc networks and related concepts.
- To understand the platforms and protocols used in mobile environment.

Course Outcomes:

- Able to think and develop new mobile application.
- Able to take any new technical issue related to this new paradigm and come up with a solution(s).
- Able to develop new ad hoc network applications and/or algorithms/protocols.
- Able to understand & develop any existing or new protocol related to mobile environment

UNIT - I

Introduction: Mobile Communications, Mobile Computing – Paradigm, Promises/Novel Applications and Impediments and Architecture; Mobile and Handheld Devices, Limitations of Mobile and Handheld Devices.

GSM – Services, System Architecture, Radio Interfaces, Protocols, Localization, Calling, Handover, Security, New Data Services, GPRS, CSHSD, DECT.

UNIT – II

(Wireless) Medium Access Control (MAC): Motivation for a specialized MAC (Hidden and exposed terminals, Near and far terminals), SDMA, FDMA, TDMA, CDMA, Wireless LAN/(IEEE 802.11)

Mobile Network Layer: IP and Mobile IP Network Layers, Packet Delivery and Handover Management, Location Management, Registration, Tunneling and Encapsulation, Route Optimization, DHCP.

UNIT – III

Mobile Transport Layer: Conventional TCP/IP Protocols, Indirect TCP, Snooping TCP, Mobile TCP, Other Transport Layer Protocols for Mobile Networks.

Database Issues: Database Hoarding & Caching Techniques, Client-Server Computing & Adaptation, Transactional Models, Query processing, Data Recovery Process & QoS Issues.

UNIT - IV

Data Dissemination and Synchronization: Communications Asymmetry, Classification of Data Delivery Mechanisms, Data Dissemination, Broadcast Models, Selective Tuning and Indexing Methods, Data Synchronization – Introduction, Software, and Protocols

UNIT - V

Mobile Adhoc Networks (MANETs): Introduction, Applications & Challenges of a MANET, Routing, Classification of Routing Algorithms, Algorithms such as DSR, AODV, DSDV, etc. , Mobile Agents, Service Discovery.

Protocols and Platforms for Mobile Computing: WAP, Bluetooth, XML, J2ME, Java Card, Palm OS, Windows CE, Symbian OS, Linux for Mobile Devices, Android.

TEXT BOOKS:

1. Jochen Schiller, “Mobile Communications”, Addison-Wesley, Second Edition, 2009.
2. Raj Kamal, “Mobile Computing”, Oxford University Press, 2007, ISBN: 0195686772.

REFERENCE BOOKS:

1. Jochen Schiller, “Mobile Communications”, Addison-Wesley, Second Edition, 2004.
2. Stojmenovic and Cacute, “Handbook of Wireless Networks and Mobile Computing”, Wiley, 2002, ISBN 0471419028.
3. Reza Behravanfar, “Mobile Computing Principles: Designing and Developing Mobile Applications with UML and XML”, ISBN: 0521817331, Cambridge University Press, Oct 2004.

DESIGN PATTERNS
(PROFESSIONAL ELECTIVE - I)

B.Tech. III Year II Sem.
Course Code: CS612PE

L	T	P	C
3	0	0	3

Prerequisites

1. A Course on Software Engineering”
2. A Course on “Object Oriented Programming Through Java”

Course Objectives:

- The aim of the course is to appreciate the idea behind Design Patterns in handling common problems faced during building an application
- This course covers all pattern types from creational to structural, behavioral to concurrency and highlights the scenarios when one pattern must be chosen over others.

Course Outcomes:

- Create software designs that are scalable and easily maintainable
- Understand the best use of Object Oriented concepts for creating truly OOP programs
- Use creational design patterns in software design for class instantiation
- Use structural design patterns for better class and object composition
- Use behavioral patterns for better organization and communication between the objects
- Use refactoring to compose the methods for proper code packaging
- Use refactoring to better organize the class responsibilities of current code

UNIT - I

Introduction: What is a design pattern? design patterns in Smalltalk MVC, Describing Design Patterns, The Catalog of Design Patterns, Organizing the Catalog, How Design Patterns Solve Design Problems, How to Select a Design Pattern, How to Use a Design Pattern.

UNIT - II

Designing a Document Editor: Design Problems, Document Structure, Formatting, Embellishing the User Interface, Supporting Multiple Look-and-Feel Standards, Supporting Multiple Window Systems, User Operations Spelling Checking and Hyphenation, Summary

UNIT - III

Creational Patterns: Abstract Factory, Builder, Factory Method, Prototype, Singleton, Discussion of Creational Patterns.

UNIT - IV

Structural Pattern: Adapter, Bridge, Composite, Decorator, Façade, Flyweight, Proxy

UNIT - V

Behavioral Patterns: Chain of Responsibility, Command, Interpreter, Iterator, Mediator, Memento, Observer, State, Strategy, Template Method, Visitor.

TEXT BOOK:

1. Design Patterns, Erich Gamma, Pearson Education

REFERENCE BOOKS:

1. Pattern's in Java, Vol –I, Mark Grand, Wiley Dream Tech.
2. Patterns in Java, Vol-II, Mark Grand, Wiley Dream Tech.
3. Java Enterprise Design Patterns Vol-III, Mark Grand, Wiley Dream Tech.
4. Head First Design Patterns, Eric Freeman, O'reily publications

ARTIFICIAL INTELLIGENCE
(Professional Elective - I)

B.Tech. III Year II Sem.
Course Code: CS613PE

L	T	P	C
3	0	0	3

Course Objectives:

- To learn the difference between optimal reasoning vs human like reasoning
- To understand the notions of state space representation, exhaustive search, heuristic search along with the time and space complexities
- To learn different knowledge representation techniques
- To understand the applications of AI: namely Game Playing, Theorem Proving, Expert Systems, Machine Learning and Natural Language Processing

Course Outcomes:

- Possess the ability to formulate an efficient problem space for a problem expressed in English.
- Possess the ability to select a search algorithm for a problem and characterize its time and space complexities.
- Possess the skill for representing knowledge using the appropriate technique
- Possess the ability to apply AI techniques to solve problems of Game Playing, Expert Systems, Machine Learning and Natural Language Processing

UNIT - I

Introduction, History, Intelligent Systems, Foundations of AI, Sub areas of AI, Applications. Problem Solving - State-Space Search and Control Strategies: Introduction, General Problem Solving, Characteristics of Problem, Exhaustive Searches, Heuristic Search Techniques, Iterative-Deepening A*, Constraint Satisfaction. Game Playing, Bounded Look-ahead Strategy and use of Evaluation Functions, Alpha-Beta Pruning

UNIT - II

Logic Concepts and Logic Programming: Introduction, Propositional Calculus, Propositional Logic, Natural Deduction System, Axiomatic System, Semantic Tableau System in Propositional Logic, Resolution Refutation in Propositional Logic, Predicate Logic, Logic Programming.

Knowledge Representation: Introduction, Approaches to Knowledge Representation, Knowledge Representation using Semantic Network, Extended Semantic Networks for KR, Knowledge Representation using Frames.

UNIT - III

Expert System and Applications: Introduction, Phases in Building Expert Systems, Expert System Architecture, Expert Systems Vs Traditional Systems, Truth Maintenance Systems, Application of Expert Systems, List of Shells and Tools.

Uncertainty Measure - Probability Theory: Introduction, Probability Theory, Bayesian Belief Networks, Certainty Factor Theory, Dempster-Shafer Theory.

UNIT-IV

Machine-Learning Paradigms: Introduction. Machine Learning Systems. Supervised and Unsupervised Learning. Inductive Learning. Learning Decision Trees (Text Book 2), Deductive Learning. Clustering, Support Vector Machines.

Artificial Neural Networks: Introduction, Artificial Neural Networks, Single-Layer Feed-Forward Networks, Multi-Layer Feed-Forward Networks, Radial-Basis Function Networks, Design Issues of Artificial Neural Networks, Recurrent Networks.

UNIT-V

Advanced Knowledge Representation Techniques: Case Grammars, Semantic Web

Natural Language Processing: Introduction, Sentence Analysis Phases, Grammars and Parsers, Types of Parsers, Semantic Analysis, Universal Networking Knowledge.

TEXT BOOKS:

1. Saroj Kaushik. Artificial Intelligence. Cengage Learning. 2011
2. Russell, Norvig: Artificial intelligence, A Modern Approach, Pearson Education, Second Edition. 2004

REFERENCE BOOK:

1. Rich, Knight, Nair: Artificial intelligence, Tata McGraw Hill, Third Edition 2009.
2. Introduction to Artificial Intelligence by Eugene Charniak, Pearson.
3. Introduction to Artificial Intelligence and expert systems Dan W.Patterson. PHI.
4. Artificial Intelligence by George Fluger Pearson fifth edition.

**INFORMATION SECURITY MANAGEMENT (SECURITY ANALYST-I)
(PROFESSIONAL ELECTIVE - I)**

B.Tech. III Year II Sem.
Course Code: CS614PE

L	T	P	C
3	0	0	3

Course Objectives:

- To introduce the terminology, technology and its applications
- To introduce the concept of Security Analyst
- To introduce the tools, technologies & programming languages which are used in day to day security analyst job role

UNIT - I

Information Security Management: Information Security Overview, Threats and Attack Vectors, Types of Attacks, Common Vulnerabilities, and Exposures (CVE), Security Attacks, Fundamentals of Information Security, Computer Security Concerns, Information Security Measures etc.

Manage your work to meet requirements (NOS 9001)

UNIT - II

Fundamentals of Information Security: Key Elements of Networks, Logical Elements of Network, Critical Information Characteristics, Information States etc.

Work effectively with Colleagues (NOS 9002)

UNIT - III

Data Leakage: What is Data Leakage and statistics, Data Leakage Threats, Reducing the Risk of Data Loss, Key Performance Indicators (KPI), Database Security etc.

UNIT - IV

Information Security Policies, Procedures, and Audits: Information Security Policies-necessity-key elements & characteristics, Security Policy Implementation, Configuration, Security Standards-Guidelines & Frameworks etc.

UNIT - V

Information Security Management – Roles and Responsibilities: Security Roles & Responsibilities, Accountability, Roles, and Responsibilities of Information Security Management, team-responding to emergency situation-risk analysis process etc.

TEXT BOOKS:

1. Management of Information Security by Michael E. Whitman and Herbert J. Mattord

REFERENCES:

1. <http://www.iso.org/iso/home/standards/management-standards/iso27001.htm>
2. <http://csrc.nist.gov/publications/nistpubs/800-55-Rev1/SP800-55-rev1.pdf>

**INTRODUCTION TO ANALYTICS (ASSOCIATE ANALYTICS -I)
(PROFESSIONAL ELECTIVE - I)**

B.Tech. III Year II Sem.
Course Code: CS615PE

L	T	P	C
3	0	0	3

Course Objectives:

- To introduce the terminology, technology and its applications
- To introduce the concept of Analytics for Business
- To introduce the tools, technologies & programming languages which is used in day to day analytics cycle

UNIT - I

Introduction to Analytics and R programming (NOS 2101): Introduction to R, R Studio (GUI): R Windows Environment, introduction to various data types, Numeric, Character, date, data frame, array, matrix etc., Reading Datasets, Working with different file types .txt, .csv etc. Outliers, Combining Datasets, R Functions and loops.

Manage your work to meet requirements (NOS 9001): Understanding Learning objectives, Introduction to work & meeting requirements, Time Management, Work management & prioritization, Quality & Standards Adherence,

UNIT - II

Summarizing Data & Revisiting Probability (NOS 2101): Summary Statistics - Summarizing data with R, Probability, Expected, Random, Bivariate Random variables, Probability distribution. Central Limit Theorem etc.

Work effectively with Colleagues (NOS 9002): Introduction to work effectively, Team Work, Professionalism, Effective Communication skills, etc.

UNIT - III

SQL using R Introduction to NoSQL, Connecting R to NoSQL databases. Excel and R integration with R connector.

UNIT - IV

Correlation and Regression Analysis (NOS 9001): Regression Analysis, Assumptions of OLS Regression, Regression Modelling. Correlation, ANOVA, Forecasting, Heteroscedasticity, Autocorrelation, Introduction to Multiple Regression etc.

UNIT - V

Understand the Verticals - Engineering, Financial and others (NOS 9002)

Understanding systems viz. Engineering Design, Manufacturing, Smart Utilities, Production lines, Automotive, Technology etc. Understanding Business problems related to various businesses.

Requirements Gathering: Gathering all the data related to Business objective.

TEXT BOOK:

1. Student's Handbook for Associate Analytics.

REFERENCE BOOKS:

1. Introduction to Probability and Statistics Using R, ISBN: 978-0-557-24979-4, is a textbook written for an undergraduate course in probability and statistics.
2. An Introduction to R, by Venables and Smith and the R Development Core Team. This may be downloaded for free from the R Project website (<http://www.r-project.org/>, see Manuals). There are plenty of other free references available from the R Project website.
3. Montgomery, Douglas C., and George C. Runger, Applied statistics and probability for engineers. John Wiley & Sons, 2010
4. Time Series Analysis and Mining with R. Yanchang Zhao.

CRYPTOGRAPHY AND NETWORK SECURITY LAB

B.Tech. III Year II Sem.
Course Code: CS604PC

L T P C
0 0 3 2

1. Write a C program that contains a string (char pointer) with a value 'Hello world'. The program should XOR each character in this string with 0 and displays the result.
2. Write a C program that contains a string (char pointer) with a value 'Hello world'. The program should AND or and XOR each character in this string with 127 and display the result.
3. Write a Java program to perform encryption and decryption using the following algorithms
 - a. Ceaser cipher
 - b. Substitution cipher
 - c. Hill Cipher
4. Write a C/JAVA program to implement the DES algorithm logic.
5. Write a C/JAVA program to implement the Blowfish algorithm logic.
6. Write a C/JAVA program to implement the Rijndael algorithm logic.
7. Write the RC4 logic in Java Using Java cryptography; encrypt the text "Hello world" using Blowfish. Create your own key using Java key tool.
8. Write a Java program to implement RSA algorithm.
9. Implement the Diffie-Hellman Key Exchange mechanism using HTML and JavaScript.
10. Calculate the message digest of a text using the SHA-1 algorithm in JAVA.
11. Calculate the message digest of a text using the MD5 algorithm in JAVA.

WEB TECHNOLOGIES LAB

B.Tech. III Year II Sem.
Course Code: CS605PC

L	T	P	C
0	0	3	2

Course Objectives:

- To enable the student to program web applications using the following technologies HTML, Javascript , AJAX, PHP, Tomcat Server, Servlets, JSP

Course Outcomes:

- Use LAMP Stack for web applications
- Use Tomcat Server for Servlets and JSPs
- Write simple applications with Technologies like HTML, Javascript, AJAX, PHP, Servlets and JSPs
- Connect to Database and get results
- Parse XML files using Java (DOM and SAX parsers)

Note:

1. Use LAMP Stack (Linux, Apache, MySQL and PHP) for the Lab Experiments. Though not mandatory, encourage the use of Eclipse platform wherever applicable
 2. The list suggests the minimum program set. Hence, the concerned staff is requested to add more problems to the list as needed
1. Install the following on the local machine
 - Apache Web Server (if not installed)
 - Tomcat Application Server locally
 - Install MySQL (if not installed)
 - Install PHP and configure it to work with Apache web server and MySQL (if not already configured)
 2. Write an HTML page including javascript that takes a given set of integer numbers and shows them after sorting in descending order.
 3. Write an HTML page including any required Javascript that takes a number from one text field in the range of 0 to 999 and shows it in another text field in words. If the number is out of range, it should show “out of range” and if it is not a number, it should show “not a number” message in the result box.
 4. Write an HTML page that has one input, which can take multi-line text and a submit button. Once the user clicks the submit button, it should show the number of characters, words and lines in the text entered using an alert message. Words are separated with white space and lines are separated with new line character.

5. Write an HTML page that contains a selection box with a list of 5 countries. When the user selects a country, its capital should be printed next to the list. Add CSS to customize the properties of the font of the capital (color, bold and font size).
6. Create an XML document that contains 10 users information. Write a Java program, which takes User Id as input and returns the user details by taking the user information from the XML document using (a) DOM Parser and (b) SAX parser

Implement the following web applications using (a) PHP, (b) Servlets and (c) JSP:

7. A user validation web application, where the user submits the login name and password to the server. The name and password are checked against the data already available in Database and if the data matches, a successful login page is returned. Otherwise a failure message is shown to the user.
8. Modify the above program to use an xml file instead of database.
9. Modify the above program to use AJAX to show the result on the same page below the submit button.
10. A simple calculator web application that takes two numbers and an operator (+, -, /, * and %) from an HTML page and returns the result page with the operation performed on the operands.
11. Modify the above program such that it stores each query in a database and checks the database first for the result. If the query is already available in the DB, it returns the value that was previously computed (from DB) or it computes the result and returns it after storing the new query and result in DB.
12. A web application takes a name as input and on submit it shows a hello <name> page where <name> is taken from the request. It shows the start time at the right top corner of the page and provides a logout button. On clicking this button, it should show a logout page with Thank You <name> message with the duration of usage (hint: Use session to store name and time).
13. A web application that takes name and age from an HTML page. If the age is less than 18, it should send a page with “Hello <name>, you are not authorized to visit this site” message, where <name> should be replaced with the entered name. Otherwise it should send “Welcome <name> to this site” message.
14. A web application for implementation:
The user is first served a login page which takes user's name and password. After submitting the details the server checks these values against the data from a database and takes the following decisions.
If name and password matches, serves a welcome page with user's full name.

If name matches and password doesn't match, then serves "password mismatch" page
If name is not found in the database, serves a registration page, where user's full name is asked and on submitting the full name, it stores, the login name, password and full name in the database (hint: use session for storing the submitted login name and password)

15. A web application that lists all cookies stored in the browser on clicking "List Cookies" button. Add cookies if necessary.

REFERENCE BOOKS:

1. The Complete Reference PHP – Steven Holzner, Tata McGraw-Hill
2. Web Programming, building internet applications, Chris Bates 2nd edition, Wiley Dreamtech
3. Java Server Pages –Hans Bergsten, SPD O'Reilly
4. Java Script, D.Flanagan, O'Reilly, SPD.
5. Internet and World Wide Web – How to program, Dietel and Nieto, Pearson.

ADVANCED ENGLISH COMMUNICATION SKILLS LAB

B.Tech. III Year II Sem.

L T P C

Course Code: EN606HS

0 0 3 2

Introduction

A course on *Advanced English Communication Skills (AECS) Lab* is considered essential at the third year level of B.Tech and B.Pharmacy courses. At this stage, the students need to prepare themselves for their career which requires them to listen to, read, speak and write in English both for their professional and interpersonal communication. The main purpose of this course is to prepare the students of Engineering for their placements.

Course Objectives: This Lab focuses on using multi-media instruction for language development to meet the following targets:

- To improve students' fluency in spoken English
- To enable them to listen to English spoken at normal conversational speed
- To help students develop their vocabulary
- To read and comprehend texts in different contexts
- To communicate their ideas relevantly and coherently in writing
- To make students industry-ready
- To help students acquire behavioral skills for their personal and professional life
- To respond appropriately in different socio-cultural and professional contexts

Course Outcomes: Students will be able to:

- Acquire vocabulary and use it contextually
- Listen and speak effectively
- Develop proficiency in academic reading and writing
- Increase possibilities of job prospects
- Communicate confidently in formal and informal contexts

Syllabus

The following course activities will be conducted as part of the Advanced English Communication Skills (AECS) Lab:

1. **Inter-personal Communication and Building Vocabulary** - Starting a Conversation – Responding Appropriately and Relevantly – Using Appropriate Body Language – Role Play in Different Situations - Synonyms and Antonyms, One-word Substitutes, Prefixes and Suffixes, Idioms and Phrases and Collocations.
2. **Reading Comprehension** –General Vs Local Comprehension, Reading for Facts, Guessing Meanings from Context, , Skimming, Scanning, Inferring Meaning.
3. **Writing Skills** – Structure and Presentation of Different Types of Writing – Letter Writing/Resume Writing/ e-correspondence/ Technical Report Writing.

4. **Presentation Skills** – Oral Presentations (individual or group) through JAM Sessions/Seminars/PPTs and Written Presentations through Posters/Projects/Reports/e-mails/Assignments... etc.,
5. **Group Discussion and Interview Skills** – Dynamics of Group Discussion, Intervention, Summarizing, Modulation of Voice, Body Language, Relevance, Fluency and Organization of Ideas and Rubrics of Evaluation- Concept and Process, Pre-interview Planning, Opening Strategies, Answering Strategies, Interview through Tele-conference & Video-conference and Mock Interviews.

Minimum Hardware Requirement: Advanced English Communication Skills (AECS) Laboratory shall have the following infrastructural facilities to accommodate at least 35 students in the lab:

- **Spacious room with appropriate acoustics**
- **Eight round tables with five movable chairs for each table.**
- **Audio-visual aids**
- **LCD Projector**
- **Public Address system**
- **Computer with suitable configuration**

Suggested Software: The software consisting of the prescribed topics elaborated above should be procured and used.

- **Oxford Advanced Learner's Compass, 8th Edition**
- **DELTA's key to the Next Generation TOEFL Test: Advanced Skill Practice.**

REFERENCES:

1. Kumar, Sanjay and Pushp Lata. English for Effective Communication, Oxford University Press, 2015.
2. Konar, Nira. English Language Laboratories – A Comprehensive Manual, PHI Learning Pvt. Ltd., 2011.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD**B.TECH. COMPUTER SCIENCE AND ENGINEERING
IV YEAR COURSE STRUCTURE & SYLLABUS (R16)****Applicable From 2016-17 Admitted Batch****IV YEAR I SEMESTER**

S. No	Course Code	Course Title	L	T	P	Credits
1	CS701PC	Data Mining	4	0	0	4
2	CS702PC	Principles of Programming Languages	4	0	0	4
3		Professional Elective – II	3	0	0	3
4		Professional Elective – III	3	0	0	3
5		Professional Elective – IV	3	0	0	3
6	CS703PC	Data Mining Lab	0	0	3	2
7		PE-II Lab #	0	0	3	2
	CS751PC	Python Programming Lab				
	CS752PC	Mobile Application Development Lab				
	CS753PC	Web Scripting Languages Lab				
	CS754PC	Internet of Things Lab				
8	CS705PC	Industry Oriented Mini Project	0	0	3	2
9	CS706PC	Seminar	0	0	2	1
		Total Credits	17	0	11	24

Courses in PE - II and PE - II Lab must be in 1-1 correspondence.

IV YEAR II SEMESTER

S. No	Course Code	Course Title	L	T	P	Credits
1		Open Elective – III	3	0	0	3
2		Professional Elective – V	3	0	0	3
3		Professional Elective – VI	3	0	0	3
4	CS801PC	Major Project	0	0	30	15
		Total Credits	9	0	30	24

Professional Elective – I

CS611PE	Mobile Computing
CS612PE	Design Patterns
CS613PE	Artificial Intelligence
CS614PE	Information Security Management (Security Analyst - I)
CS615PE	Introduction to Analytics (Associate Analytics - I)

Professional Elective – II

CS721PE	Python Programming
CS722PE	Mobile Application Development
CS723PE	Web Scripting Languages
CS724PE	Internet of Things

Professional Elective - III

CS731PE	Graph Theory
CS732PE	Distributed Systems
CS733PE	Machine Learning
CS734PE	Software Process and Project Management

Professional Elective - IV

CS741PE	Computational Complexity
CS742PE	Cloud Computing
CS743PE	Blockchain Technology
CS744PE	Social Network Analysis

Professional Elective - V

CS851PE	Information Theory & Coding
CS852PE	Real-Time Systems
CS853PE	Data Analytics
CS854PE	Modern Software Engineering

Professional Elective - VI

CS861PE	Advanced Algorithms
CS862PE	Web Services and Service Oriented Architecture
CS863PE	Computer Forensics
CS864PE	Neural Networks and Deep Learning

***Open Elective** subjects' syllabus is provided in a separate document.

***Open Elective** – Students should take Open Electives from the List of Open Electives Offered by Other Departments/Branches Only.

Ex: - A Student of Mechanical Engineering can take Open Electives from all other departments/branches except Open Electives offered by Mechanical Engineering Dept.

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD
LIST OF OPEN ELECTIVES OFFERED BY VARIOUS DEPARTMENTS FOR
B.TECH. III AND IV YEARS

S. No.	Name of the Department Offering Open Electives	Open Elective – I (Semester – V)	Open Elective – II (Semester – VI)
1	Aeronautical Engg.	AE511OE: Introduction to Space Technology	AE621OE: Introduction to Aerospace Engineering
2	Automobile Engg.	CE511OE: Disaster Management MT512OE: Intellectual Property Rights	MT621OE: Data Structures MT622OE: Artificial Neural Networks
3	Biomedical Engg.	BM511OE: Reliability Engineering	BM621OE: Medical Electronics
4	Civil Engg.	CE511OE: Disaster Management.	CE621OE: Remote Sensing and GIS CE622OE: Geo-Informatics CE623OE: Intellectual Property Rights
5	Civil and Environmental Engg.	CE511OE: Disaster Management	CN621OE: Environmental Impact Assessment CE623OE: Intellectual Property Rights
6	Computer Science and Engg. / Information Technology	CS511OE: Operating Systems CS512OE: Database Management Systems	CS621OE: Java Programming CS622OE: Software Testing Methodologies CS623OE: Cyber Security
7	Electronics and Communication Engg. / Electronics and Telematics Engg.	EC511OE: Principles of Electronic Communications	EC621OE: Principles of Computer Communications and Networks
8	Electronics and Computer Engg.	EM511OE: Scripting Languages	EM621OE: Soft Computing Techniques
9	Electrical and Electronics Engg.	EE511OE: Non-Conventional Power Generation EE512OE: Electrical Engineering Materials EE513OE: Nanotechnology	EE621OE: Design Estimation and Costing of Electrical Systems EE622OE: Energy Storage Systems EE623OE: Introduction to Mechatronics
10	Electronics and Instrumentation Engg.	EI511OE: Electronic Measurements and Instrumentation	EI621OE: Industrial Electronics
11	Mechanical Engg.	ME511OE: Optimization Techniques ME512OE: Computer Graphics ME513OE: Introduction	ME621OE: World Class Manufacturing ME622OE: Fundamentals of Robotics ME623OE: Fabrication

		to Mechatronics ME514OE: Fundamentals of Mechanical Engineering	Processes
12	Mechanical Engg. (Material Science and Nanotechnology)	NT511OE: Fabrication Processes NT512OE: Nondestructive Testing Methods NT513OE: Fundamentals of Engineering Materials	NT621OE: Introduction to Material Handling NT622OE: Non-Conventional Energy Sources NT623OE: Robotics
13	Mechanical Engg. (mechatronics)	MT511OE: Analog and Digital I.C. Applications MT512OE: Intellectual Property Rights MT513OE: Computer Organization	MT621OE: Data Structures MT622OE: Artificial Neural Networks MT623OE: Industrial Management
14	Metallurgical and Materials Engg.	MM511OE: Materials Characterization Techniques	MM621OE: Science and Technology of Nano Materials MM622OE: Metallurgy of Non Metallurgists
15	Mining Engg.	MN511OE: Introduction to Mining Technology	MN621OE: Coal Gasification, Coal Bed Methane and Shale Gas
16	Petroleum Engg.	PE511OE: Materials Science and Engineering PE512OE: Renewable Energy Sources PE513OE: Environmental Engineering	PE621OE: Energy Management and Conservation PE622OE: Optimization Techniques PE623OE: Entrepreneurship and Small Business Enterprises

S. No.	Name of the Department Offering Open Electives	Open Elective –III (Semester – VIII)
1	Aeronautical Engg.	AE831OE: Air Transportation Systems AE832OE: Rockets and Missiles
2	Automobile Engg.	AM831OE: Introduction to Mechatronics AM832OE: Microprocessors and Microcontrollers
3	Biomedical Engg.	BM831OE: Telemetry and Telecontrol BM832OE: Electromagnetic Interference and Compatibility
4	Civil Engg.	CE831OE: Environmental Impact Assessment CE832OE: Optimization Techniques in Engineering CE833OE: Entrepreneurship and Small Business Enterprises
5	Civil and Environmental Engg.	CN831OE: Remote Sensing and GIS CE833OE: Entrepreneurship and Small Business

		Enterprises
6	Computer Science and Engg. / Information Technology	CS831OE: Linux Programming CS832OE: R Programming CS833OE: PHP Programming
7	Electronics and Communication Engg. / Electronics and Telematics Engg.	EC831OE: Electronic Measuring Instruments
8	Electronics and Computer Engg.	EM831OE: Data Analytics
9	Electrical and Electronics Engg.	EE831OE: Entrepreneur Resource Planning EE832OE: Management Information Systems EE833OE: Organizational Behaviour
10	Electronics and Instrumentation Engg.	EI831OE: Sensors and Transducers, EI832OE: PC Based Instrumentation
11	Mechanical Engg.	ME831OE: Total Quality Management ME832OE: Industrial Safety, Health, and Environmental Engineering ME833OE: Basics of Thermodynamics ME834OE: Reliability Engineering
12	Mechanical Engg. (Material Science and Nanotechnology)	NT831OE: Concepts of Nano Science And Technology NT832OE: Synthesis of Nanomaterials NT833OE: Characterization of Nanomaterials
13	Mechanical Engg. (mechatronics)	MT831OE: Renewable Energy Sources MT832OE: Production Planning and Control CE833OE: Entrepreneurship and Small Business Enterprises
14	Metallurgical and Materials Engg.	MM831OE: Design and Selection of Engineering Materials
15	Mining Engg.	MN831OE: Solid Fuel Technology MN832OE: Health & Safety in Mines
16	Petroleum Engg.	PE831OE: Disaster Management PE832OE: Fundamentals of Liquefied Natural Gas PE833OE: Health, Safety and Environment in Petroleum Industry

***Open Elective** – Students should take Open Electives from List of Open Electives Offered by Other Departments/Branches Only.

Ex: - A Student of Mechanical Engineering can take Open Electives from all other departments/branches except Open Electives offered by Mechanical Engineering Dept.

DATA MINING

B.Tech. IV Year I Sem.
Course Code: CS701PC

L T P C
4 0 0 4

Course Objectives:

- Learn data mining concepts understand association rules mining.
- Discuss classification algorithms learn how data is grouped using clustering techniques.
- To develop the abilities of critical analysis to data mining systems and applications.
- To implement practical and theoretical understanding of the technologies for data mining
- To understand the strengths and limitations of various data mining models;

Course Outcomes:

- Ability to perform the preprocessing of data and apply mining techniques on it.
- Ability to identify the association rules, classification and clusters in large data sets.
- Ability to solve real world problems in business and scientific information using data mining
- Ability to classify web pages, extracting knowledge from the web

UNIT - I

Introduction to Data Mining: Introduction, What is Data Mining, Definition, KDD, Challenges, Data Mining Tasks, Data Preprocessing, Data Cleaning, Missing data, Dimensionality Reduction, Feature Subset Selection, Discretization and Binaryzation, Data Transformation; Measures of Similarity and Dissimilarity- Basics.

UNIT - II

Association Rules: Problem Definition, Frequent Item Set Generation, The APRIORI Principle, Support and Confidence Measures, Association Rule Generation; APRIORI Algorithm, The Partition Algorithms, FP-Growth Algorithms, Compact Representation of Frequent Item Set- Maximal Frequent Item Set, Closed Frequent Item Set.

UNIT - III

Classification: Problem Definition, General Approaches to solving a classification problem , Evaluation of Classifiers , Classification techniques, Decision Trees-Decision tree Construction , Methods for Expressing attribute test conditions, Measures for Selecting the Best Split, Algorithm for Decision tree Induction ; Naive-Bayes Classifier, Bayesian Belief Networks; K- Nearest neighbor classification-Algorithm and Characteristics.

UNIT - IV

Clustering: Problem Definition, Clustering Overview, Evaluation of Clustering Algorithms, Partitioning Clustering-K-Means Algorithm, K-Means Additional issues, PAM Algorithm;

Hierarchical Clustering-Agglomerative Methods and divisive methods, Basic Agglomerative Hierarchical Clustering Algorithm, Specific techniques, Key Issues in Hierarchical Clustering, Strengths and Weakness; Outlier Detection.

UNIT - V

Web and Text Mining: Introduction, web mining, web content mining, web structure mining, we usage mining, Text mining –unstructured text, episode rule discovery for texts, hierarchy of categories, text clustering.

TEXT BOOKS:

1. Data Mining- Concepts and Techniques- Jiawei Han, Micheline Kamber, Morgan Kaufmann Publishers, Elsevier, 2 Edition, 2006.
2. Introduction to Data Mining, Pang-Ning Tan, Vipin Kumar, Michael Steinbanch, Pearson Education.
3. Data mining Techniques and Applications, Hongbo Du Cengage India Publishing

REFERENCE BOOKS:

1. Data Mining Techniques, Arun K Pujari, 3rd Edition, Universities Press.
2. Data Mining Principles & Applications – T.V Sveresh Kumar, B.Esware Reddy, Jagadish S Kalimani, Elsevier.
3. Data Mining, Vikaram Pudi, P Radha Krishna, Oxford University Press

PRINCIPLES OF PROGRAMMING LANGUAGES

B.Tech. IV Year I Sem.
Course Code: CS702PC

L	T	P	C
4	0	0	4

Course Objectives:

- To introduce the various programming paradigms.
- To understand the evolution of programming languages.
- To understand the concepts of OO languages, functional languages, logical and scripting languages.
- To introduce the principles and techniques involved in design and implementation of modern programming languages.
- To introduce the notations to describe the syntax and semantics of programming languages.
- To introduce the concepts of concurrency control and exception handling.
- To introduce the concepts of ADT and OOP for software development.

Course Outcomes:

- Ability to express syntax and semantics in formal notation.
- Ability to apply suitable programming paradigm for the application.
- Ability to compare the features of various programming languages.
- Able to understand the programming paradigms of modern programming languages.
- Able to understand the concepts of ADT and OOP.
- Ability to program in different language paradigms and evaluate their relative benefits.

UNIT-I

Preliminary Concepts: Reasons for studying concepts of programming languages, programming domains, language evaluation criteria, influences on language design, language categories, language design trade-offs, implementation methods, programming environments, Evolution of Major Programming Languages.

Syntax and Semantics: General problem of describing syntax, formal methods of describing syntax, attribute grammars, describing the meanings of programs

UNIT-II

Names, Bindings, and Scopes: Introduction, names, variables, concept of binding, scope, scope and lifetime, referencing environments, named constants

Data types: Introduction, primitive, character, string types, user defined ordinal types, array, associative arrays, record, tuple types, list types, union types, pointer and reference types, type checking, strong typing, type equivalence

Expressions and Statements: Arithmetic expressions, overloaded operators, type conversions, relational and boolean expressions, short- circuit evaluation, assignment statements, mixed-mode assignment

Control Structures – introduction, selection statements, iterative statements, unconditional branching, guarded commands.

UNIT-III

Subprograms: Fundamentals of subprograms, design issues for subprograms, local referencing environments, parameter passing methods, parameters that are subprograms, calling subprograms indirectly, overloaded subprograms, generic subprograms, design issues for functions, user defined overloaded operators, closures, co routines

Implementing subprograms: General semantics of calls and returns, implementing simple subprograms, implementing subprograms with stack-dynamic local variables, nested subprograms, blocks, implementing dynamic scoping

Abstract Data types: The concept of abstraction, introductions to data abstraction, design issues, language examples, parameterized ADT, encapsulation constructs, naming encapsulations

UNIT-IV

Object Oriented Programming: Design issues for OOP, OOP in Smalltalk, C++, Java, Ada 95, Ruby, Implementation of Object-Oriented constructs.

Concurrency: introduction, introduction to subprogram level concurrency, semaphores, monitors, message passing, Ada support for concurrency, Java threads, concurrency in functional languages, statement level concurrency.

Exception Handling and Event Handling: Introduction, exception handling in Ada, C++, Java, introduction to event handling, event handling with Java and C#.

UNIT-V

Functional Programming Languages: Introduction, mathematical functions, fundamentals of functional programming language, LISP, support for functional programming in primarily imperative languages, comparison of functional and imperative languages

Logic Programming Language: Introduction, an overview of logic programming, basic elements of prolog, deficiencies of prolog, applications of logic programming.

Scripting Language: Pragmatics, Key Concepts, Case Study: Python – Values and Types, Variables, Storage and Control, Bindings and Scope, Procedural Abstraction, Data Abstraction, Separate Compilation, Module Library. (Text Book 2)

TEXT BOOKS:

1. Concepts of Programming Languages, Robert .W. Sebesta 10th edition, Pearson Education.
2. Programming Language Design Concepts, D. A. Watt, Wiley India Edition.

REFERENCE BOOK:

1. Programming Languages, A.B. Tucker, R.E. Noonan, TMH.
2. Programming Languages, K. C. Louden and K A Lambert., 3rd edition, Cengage Learning.
3. Programming Language Concepts, C Ghezzi and M Jazayeri, Wiley India.
4. Programming Languages 2nd Edition Ravi Sethi Pearson.
5. Introduction to Programming Languages Arvind Kumar Bansal CRC Press.

PYTHON PROGRAMMING
(PROFESSIONAL ELECTIVE –II)

B.Tech. IV Year I Sem.
Course Code: CS721PE

L T P C
3 0 0 3

Course Objectives: This course will enable students to

- Learn Syntax and Semantics and create Functions in Python.
- Handle Strings and Files in Python.
- Understand Lists, Dictionaries and Regular expressions in Python.
- Implement Object Oriented Programming concepts in Python.
- Build Web Services and introduction to Network and Database Programming in Python.

Course Outcomes: The students should be able to:

- Examine Python syntax and semantics and be fluent in the use of Python flow control and functions.
- Demonstrate proficiency in handling Strings and File Systems.
- Create, run and manipulate Python Programs using core data structures like Lists, Dictionaries and use Regular Expressions.
- Interpret the concepts of Object-Oriented Programming as used in Python.
- Implement exemplary applications related to Network Programming, Web Services and Databases in Python.

UNIT - I

Python Basics, Objects- Python Objects, Standard Types, Other Built-in Types, Internal Types, Standard Type Operators, Standard Type Built-in Functions, Categorizing the Standard Types, Unsupported Types

Numbers - Introduction to Numbers, Integers, Floating Point Real Numbers, Complex Numbers, Operators, Built-in Functions, Related Modules

Sequences - Strings, Lists, and Tuples, Mapping and Set Types

UNIT - II

FILES: File Objects, File Built-in Function [open()], File Built-in Methods, File Built-in Attributes, Standard Files, Command-line Arguments, File System, File Execution, Persistent Storage Modules, Related Modules

Exceptions: Exceptions in Python, Detecting and Handling Exceptions, Context Management, *Exceptions as Strings, Raising Exceptions, Assertions, Standard Exceptions, *Creating Exceptions, Why Exceptions (Now)?, Why Exceptions at All?, Exceptions and the sys Module, Related Modules

Modules: Modules and Files, Namespaces, Importing Modules, Importing Module Attributes, Module Built-in Functions, Packages, Other Features of Modules

UNIT - III

Regular Expressions: Introduction, Special Symbols and Characters, Res and Python
Multithreaded Programming: Introduction, Threads and Processes, Python, Threads, and the
Global Interpreter Lock, Thread Module, Threading Module, Related Modules

UNIT - IV

GUI Programming: Introduction, Tkinter and Python Programming, Brief Tour of Other
GUIs, Related Modules and Other GUIs

WEB Programming: Introduction, Web Surfing with Python, Creating Simple Web Clients,
Advanced Web Clients, CGI-Helping Servers Process Client Data, Building CGI Application
Advanced CGI, Web (HTTP) Servers

UNIT – V

Database Programming: Introduction, Python Database Application Programmer's Interface
(DB-API), Object Relational Managers (ORMs), Related Modules

Textbook

1. Core Python Programming, Wesley J. Chun, Second Edition, Pearson.

**MOBILE APPLICATION DEVELOPMENT
(PROFESSIONAL ELECTIVE –II)**

B.Tech. IV Year I Sem.
Course Code: CS722PE

L T P C
3 0 0 3

Course Objectives:

- To demonstrate their understanding of the fundamentals of Android operating systems
- To demonstrate their skills of using Android software development tools
- To demonstrate their ability to develop software with reasonable complexity on mobile platform
- To demonstrate their ability to deploy software to mobile devices
- To demonstrate their ability to debug programs running on mobile devices

UNIT - I

Introduction to Android Operating System: Android OS design and Features – Android development framework, SDK features, Installing and running applications on Eclipse platform, Creating AVDs, Types of Android applications, Best practices in Android programming, Android tools

Android application components – Android Manifest file, Externalizing resources like values, themes, layouts, Menus etc, Resources for different devices and languages, Runtime Configuration Changes

Android Application Lifecycle – Activities, Activity lifecycle, activity states, monitoring state changes

UNIT - II

Android User Interface: Measurements – Device and pixel density independent measuring units. **Layouts** – Linear, Relative, Grid and Table Layouts.

User Interface (UI) Components – Editable and non-editable Text Views, Buttons, Radio and Toggle Buttons, Checkboxes, Spinners, Dialog and pickers.

Event Handling – Handling clicks or changes of various UI components.

Fragments – Creating fragments, Lifecycle of fragments, Fragment states, Adding fragments to Activity, adding, removing and replacing fragments with fragment transactions, interfacing between fragments and Activities, Multi-screen Activities

UNIT - III

Intents and Broadcasts: Intent – Using intents to launch Activities, Explicitly starting new Activity, Implicit Intents, Passing data to Intents, Getting results from Activities, Native Actions, using Intent to dial a number or to send SMS

Broadcast Receivers – Using Intent filters to service implicit Intents, Resolving Intent filters, finding and using Intents received within an Activity

Notifications – Creating and Displaying notifications, Displaying Toasts

UNIT - IV

Persistent Storage: Files – Using application specific folders and files, creating files, reading data from files, listing contents of a directory Shared Preferences – Creating shared preferences, saving and retrieving data using Shared Preference

Database – Introduction to SQLite database, creating and opening a database, creating tables, inserting retrieving and deleting data, Registering Content Providers, Using content Providers (insert, delete, retrieve and update)

UNIT - V

Advanced Topics: Alarms – Creating and using alarms.

Using Internet Resources – Connecting to internet resource, using download manager

Location Based Services – Finding Current Location and showing location on the Map, updating location

TEXT BOOKS:

1. Professional Android 4 Application Development, Reto Meier, Wiley India, (Wrox) , 2012
2. Android Application Development for Java Programmers, James C Sheusi, Cengage Learning, 2013

REFERENCES:

1. Beginning Android 4 Application Development, Wei-Meng Lee, Wiley India (Wrox), 2013

**WEB SCRIPTING LANGUAGES
(PROFESSIONAL ELECTIVE –II)**

B.Tech. IV Year I Sem.
Course Code: CS723PE

L	T	P	C
3	0	0	3

Prerequisites:

- A course on “Computer Programming and Data Structures.”
- A course on “Object Oriented Programming Concepts.”

Course Objectives:

- This course introduces the script programming paradigm.
- Introduces scripting languages such as Perl, Ruby and TCL.
- Learning TCL.

Course Outcomes:

- Comprehend the differences between typical scripting languages and typical system and application programming languages.
- Gain knowledge of the strengths and weakness of Perl, TCL and Ruby; and select an appropriate language for solving a given problem.
- Acquire programming skills in scripting language.

UNIT - I

Introduction: Ruby, Rails, the structure and Execution of Ruby Programs, Package Management with RUBYGEMS, Ruby and web: Writing CGI scripts, cookies, Choice of Webservers, SOAP and webservices

RubyTk – Simple Tk Application, widgets, Binding events, Canvas, scrolling

UNIT - II

Extending Ruby: Ruby Objects in C, the Jukebox extension, Memory allocation, Ruby Type System, Embedding Ruby to Other Languages, Embedding a Ruby Interpreter

UNIT - III

Introduction to PERL and Scripting

Scripts and Programs, Origin of Scripting, Scripting Today, Characteristics of Scripting Languages, Uses for Scripting Languages, Web Scripting, and the universe of Scripting Languages. PERL- Names and Values, Variables, Scalar Expressions, Control Structures, arrays, list, hashes, strings, pattern and regular expressions, subroutines.

UNIT - IV

Advanced Perl

Finer points of looping, pack and unpack, filesystem, eval, data structures, packages, modules, objects, interfacing to the operating system, Creating Internet ware applications, Dirty Hands Internet Programming, security Issues.

UNIT - V

TCL: TCL Structure, syntax, Variables and Data in TCL, Control Flow, Data Structures, input/output, procedures, strings, patterns, files, Advance TCL- eval, source, exec and uplevel commands, Name spaces, trapping errors, event driven programs, making applications internet aware, Nuts and Bolts Internet Programming, Security Issues, C Interface.

TK: TK-Visual Tool Kits, Fundamental Concepts of TK, TK by example, Events and Binding, Perl-TK.

TEXT BOOKS:

1. The World of Scripting Languages, David Barron, Wiley Publications.
2. Ruby Programming language by David Flanagan and Yukihiro Matsumoto O'Reilly
3. "Programming Ruby" The Pramatic Progammmers guide by Dabve Thomas Second edition

REFERENCE BOOKS:

1. Open Source Web Development with LAMP using Linux Apache, MySQL, Perl and PHP, J.Lee and B. Ware (Addison Wesley) Pearson Education.
2. Perl by Example, E. Quigley, Pearson Education.
3. Programming Perl, Larry Wall, T. Christiansen and J. Orwant, O'Reilly, SPD.
4. Tcl and the Tk Tool kit, Ousterhout, Pearson Education.
5. Perl Power, J.P. Flynt, Cengage Learning.

INTERNET OF THINGS
(PROFESSIONAL ELECTIVE – II)

B.Tech. IV Year I Sem.

L T P C

Course Code: CS724PE/EC732PE

3 0 0 3

Course Objectives:

- To introduce the terminology, technology and its applications
- To introduce the concept of M2M (machine to machine) with necessary protocols
- To introduce the Python Scripting Language which is used in many IoT devices
- To introduce the Raspberry PI platform, that is widely used in IoT applications
- To introduce the implementation of web based services on IoT devices.

Course Outcomes:

- Interpret the impact and challenges posed by IoT networks leading to new architectural models.
- Compare and contrast the deployment of smart objects and the technologies to connect them to network.
- Appraise the role of IoT protocols for efficient network communication.
- Elaborate the need for Data Analytics and Security in IoT.
- Illustrate different sensor technologies for sensing real world entities and identify the applications of IoT in Industry.

UNIT - I

Introduction to Internet of Things –Definition and Characteristics of IoT, Physical Design of IoT – IoT Protocols, IoT communication models, Iot Communication APIs, IoT enabled Technologies – Wireless Sensor Networks, Cloud Computing, Big data analytics, Communication protocols, Embedded Systems, IoT Levels and Templates, Domain Specific IoTs – Home, City, Environment, Energy, Retail, Logistics, Agriculture, Industry, health and Lifestyle.

UNIT - II

IoT and M2M – Software defined networks, network function virtualization, difference between SDN and NFV for IoT. Basics of IoT System Management with NETCOZF, YANG- NETCONF, YANG, SNMP NETOPEER

UNIT - III

Introduction to Python - Language features of Python, Data types, data structures, Control of flow, functions, modules, packaging, file handling, data/time operations, classes, Exception handling. Python packages - JSON, XML, HTTP Lib, URL Lib, SMTP Lib.

UNIT - IV

IoT Physical Devices and Endpoints - Introduction to Raspberry PI - Interfaces (serial, SPI, I2C). Programming – Python program with Raspberry PI with focus of interfacing external gadgets, controlling output, reading input from pins.

UNIT - V

IoT Physical Servers and Cloud Offerings – Introduction to Cloud Storage models and communication APIs. Webservice – Web server for IoT, Cloud for IoT, Python web application framework. Designing a RESTful web API

TEXT BOOKS:

1. Internet of Things - A Hands-on Approach, Arshdeep Bahga and Vijay Madisetti, Universities Press, 2015, ISBN: 9788173719547
2. Getting Started with Raspberry Pi, Matt Richardson & Shawn Wallace, O'Reilly (SPD), 2014, ISBN: 9789350239759

GRAPH THEORY
(PROFESSIONAL ELECTIVE – III)

B.Tech. IV Year I Sem.
Course Code: CS731PE

L	T	P	C
3	0	0	3

Prerequisite:

1. An understanding of Mathematics in general is sufficient.

Course Outcomes:

- Know some important classes of graph theoretic problems;
- Be able to formulate and prove central theorems about trees, matching, connectivity, colouring and planar graphs;
- Be able to describe and apply some basic algorithms for graphs;
- Be able to use graph theory as a modelling tool.

UNIT - I

Introduction-Discovery of graphs, Definitions, Subgraphs, Isomorphic graphs, Matrix representations of graphs, Degree of a vertex, Directed walks, paths and cycles, Connectivity in digraphs, Eulerian and Hamilton digraphs, Eulerian digraphs, Hamilton digraphs, Special graphs, Complements, Larger graphs from smaller graphs, Union, Sum, Cartesian Product, Composition, Graphic sequences, Graph theoretic model of the LAN problem, Havel-Hakimi criterion, Realization of a graphic sequence.

UNIT - II

Connected graphs and shortest paths - Walks, trails, paths, cycles, connected graphs, Distance, Cut-vertices and cut-edges, Blocks, Connectivity, Weighted graphs and shortest paths, Weighted graphs, Dijkstra's shortest path algorithm, Floyd-Warshall shortest path algorithm.

UNIT - III

Trees- Definitions and characterizations, Number of trees, Cayley's formula, Kircho-matrix-tree theorem, Minimum spanning trees, Kruskal's algorithm, Prim's algorithm, Special classes of graphs, Bipartite Graphs, Line Graphs, Chordal Graphs, Eulerian Graphs, Fleury's algorithm, Chinese Postman problem, Hamilton Graphs, Introduction, Necessary conditions and sufficient conditions.

UNIT - IV

Independent sets coverings and matchings– Introduction, Independent sets and coverings: basic equations, Matchings in bipartite graphs, Hall's Theorem, Konig's Theorem, Perfect matchings in graphs, Greedy and approximation algorithms.

UNIT - V

Vertex Colorings- Basic definitions, Cliques and chromatic number, Mycielski's theorem, Greedy coloring algorithm, Coloring of chordal graphs, Brooks theorem, **Edge Colorings**, Introduction and Basics, Gupta-Vizing theorem, Class-1 and Class-2 graphs, Edge-coloring of bipartite graphs, Class-2 graphs, Hajos union and Class-2 graphs, A scheduling problem and equitable edge-coloring.

TEXTBOOKS:

1. J. A. Bondy and U. S. R. Murty. Graph Theory, volume 244 of Graduate Texts in Mathematics. Springer, 1st edition, 2008.
2. J. A. Bondy and U. S. R. Murty. Graph Theory with Applications.

REFERENCES:

1. Lecture Videos: <http://nptel.ac.in/courses/111106050/13>
2. Introduction To Graph Theory, Douglas B. West, Pearson.
3. Schaum's Outlines Graph Theory, Balakrishnan, TMH
4. Introduction to Graph Theory, Wilson Robin j, PHI
5. Graph Theory with Applications to Engineering and Computer Science, Narsing Deo, PHI
6. Graphs - An Introductory Approach, Wilson and Watkins

DISTRIBUTED SYSTEMS
(PROFESSIONAL ELECTIVE – III)

B.Tech. IV Year I Sem.
Course Code: CS732PE

L T P C
3 0 0 3

Course Objectives:

- To understand what and why a distributed system is.
- To understand theoretical concepts, namely, virtual time, agreement and consensus protocols.
- To understand IPC, Group Communication & RPC Concepts.
- To understand the DFS and DSM Concepts.
- To understand the concepts of transaction in distributed environment and associated concepts, namely, concurrency control, deadlocks and error recovery.

Course Outcomes:

- Able to comprehend and design a new distributed system with the desired features.
- Able to start literature survey leading to further research in any subarea.
- Able to develop new distributed applications.

UNIT- I

Characterization of Distributed Systems: Introduction, Examples of Distributed Systems, Resource Sharing and the Web, Challenges. **System Models:** Introduction, Architectural Models, Fundamental Models.

UNIT- II

Time and Global States: Introduction, Clocks Events and Process States, Synchronizing Physical Clocks, Logical Time and Logical Clocks, Global States, Distributed Debugging. **Coordination and Agreement:** Introduction, Distributed Mutual Exclusion, Elections, Multicast Communication, Consensus and Related Problems.

UNIT- III

Inter Process Communication: Introduction, The API for the Internet Protocols, External Data Representation and Marshalling, Client-Server Communication, Group Communication, Case Study: IPC in UNIX. **Distributed Objects and Remote Invocation:** Introduction, Communication between Distributed Objects, Remote Procedure Call, Events and Notifications, Case Study: JAVA RMI.

UNIT- IV

Distributed File Systems: Introduction, File Service Architecture, Case Study 1: Sun Network File System, Case Study 2: The Andrew File System.

Name Services: Introduction, Name Services and the Domain Name System, Directory Services, Case Study of the Global Name Services.

Distributed Shared Memory: Introduction, Design and Implementation Issues, Sequential Consistency and IVY case study, Release Consistency, Munin Case Study, Other Consistency Models.

UNIT- V

Transactions and Concurrency Control: Introduction, Transactions, Nested Transactions, Locks, Optimistic Concurrency Control, Timestamp Ordering, Comparison of Methods for Concurrency Control. **Distributed Transactions:** Introduction, Flat and Nested Distributed Transactions, Atomic Commit Protocols, Concurrency Control in Distributed Transactions, Distributed Deadlocks, Transaction Recovery.

TEXT BOOK:

1. Distributed Systems, Concepts and Design, George Coulouris, J Dollimore and Tim Kindberg, Pearson Education, 4th Edition, 2009.

REFERENCE BOOKS:

1. Distributed Systems, Principles and Paradigms, Andrew S. Tanenbaum, Maarten Van Steen, 2nd Edition, PHI.
2. Distributed Systems, An Algorithm Approach, Sukumar Ghosh, Chapman & Hall/CRC, Taylor & Fransis Group, 2007.

MACHINE LEARNING
(PROFESSIONAL ELECTIVE – III)

B.Tech. IV Year I Sem.
Course Code: CS733PE

L	T	P	C
3	0	0	3

Prerequisites

- Data Structures
- Knowledge on statistical methods

Course Objectives:

- This course explains machine learning techniques such as decision tree learning, Bayesian learning etc.
- To understand computational learning theory.
- To study the pattern comparison techniques.

Course Outcomes:

- Understand the concepts of computational intelligence like machine learning
- Ability to get the skill to apply machine learning techniques to address the real time problems in different areas
- Understand the Neural Networks and its usage in machine learning application.

UNIT - I

Introduction - Well-posed learning problems, designing a learning system, Perspectives and issues in machine learning

Concept learning and the general to specific ordering – introduction, a concept learning task, concept learning as search, find-S: finding a maximally specific hypothesis, version spaces and the candidate elimination algorithm, remarks on version spaces and candidate elimination, inductive bias.

Decision Tree Learning – Introduction, decision tree representation, appropriate problems for decision tree learning, the basic decision tree learning algorithm, hypothesis space search in decision tree learning, inductive bias in decision tree learning, issues in decision tree learning.

UNIT - II

Artificial Neural Networks-1– Introduction, neural network representation, appropriate problems for neural network learning, perceptions, multilayer networks and the back-propagation algorithm.

Artificial Neural Networks-2- Remarks on the Back-Propagation algorithm, An illustrative example: face recognition, advanced topics in artificial neural networks.

Evaluation Hypotheses – Motivation, estimation hypothesis accuracy, basics of sampling theory, a general approach for deriving confidence intervals, difference in error of two hypotheses, comparing learning algorithms.

UNIT - III

Bayesian learning – Introduction, Bayes theorem, Bayes theorem and concept learning, Maximum Likelihood and least squared error hypotheses, maximum likelihood hypotheses for predicting probabilities, minimum description length principle, Bayes optimal classifier, Gibbs algorithm, Naïve Bayes classifier, an example: learning to classify text, Bayesian belief networks, the EM algorithm.

Computational learning theory – Introduction, probably learning an approximately correct hypothesis, sample complexity for finite hypothesis space, sample complexity for infinite hypothesis spaces, the mistake bound model of learning.

Instance-Based Learning- Introduction, k -nearest neighbour algorithm, locally weighted regression, radial basis functions, case-based reasoning, remarks on lazy and eager learning.

UNIT- IV

Genetic Algorithms – Motivation, Genetic algorithms, an illustrative example, hypothesis space search, genetic programming, models of evolution and learning, parallelizing genetic algorithms.

Learning Sets of Rules – Introduction, sequential covering algorithms, learning rule sets: summary, learning First-Order rules, learning sets of First-Order rules: FOIL, Induction as inverted deduction, inverting resolution.

Reinforcement Learning – Introduction, the learning task, Q-learning, non-deterministic, rewards and actions, temporal difference learning, generalizing from examples, relationship to dynamic programming.

UNIT - V

Analytical Learning-1- Introduction, learning with perfect domain theories: PROLOG-EBG, remarks on explanation-based learning, explanation-based learning of search control knowledge.

Analytical Learning-2-Using prior knowledge to alter the search objective, using prior knowledge to augment search operators.

Combining Inductive and Analytical Learning – Motivation, inductive-analytical approaches to learning, using prior knowledge to initialize the hypothesis.

TEXT BOOK:

1. Machine Learning – Tom M. Mitchell, - MGH

REFERENCE:

1. Machine Learning: An Algorithmic Perspective, Stephen Marshland, Taylor & Francis

**SOFTWARE PROCESS AND PROJECT MANAGEMENT
(PROFESSIONAL ELECTIVE – III)**

B.Tech. IV Year I Sem.
Course Code: CS734PE

L T P C
3 0 0 3

Course Objectives:

- To acquire knowledge on software process management
- To acquire managerial skills for software project development
- To understand software economics

Course Outcomes:

- Gain knowledge of software economics, phases in the life cycle of software development, project organization, project control and process instrumentation
- Analyze the major and minor milestones, artifacts and metrics from management and technical perspective
- Design and develop software product using conventional and modern principles of software project management

UNIT - I

Software Process Maturity

Software maturity Framework, Principles of Software Process Change, Software Process Assessment, The Initial Process, The Repeatable Process, The Defined Process, The Managed Process, The Optimizing Process.

Process Reference Models

Capability Maturity Model (CMM), CMMI, PCMM, PSP, TSP).

UNIT - II

Software Project Management Renaissance

Conventional Software Management, Evolution of Software Economics, Improving Software Economics, The old way and the new way.

Life-Cycle Phases and Process artifacts

Engineering and Production stages, inception phase, elaboration phase, construction phase, transition phase, artifact sets, management artifacts, engineering artifacts and pragmatic artifacts, model-based software architectures.

UNIT - III

Workflows and Checkpoints of process

Software process workflows, Iteration workflows, Major milestones, minor milestones, periodic status assessments.

Process Planning

Work breakdown structures, Planning guidelines, cost and schedule estimating process, iteration planning process, Pragmatic planning.

UNIT - IV

Project Organizations

Line-of- business organizations, project organizations, evolution of organizations, process automation.

Project Control and process instrumentation

The seven-core metrics, management indicators, quality indicators, life-cycle expectations, Pragmatic software metrics, metrics automation.

UNIT - V

CCPDS-R Case Study and Future Software Project Management Practices

Modern Project Profiles, Next-Generation software Economics, Modern Process Transitions.

TEXT BOOKS:

1. Managing the Software Process, Watts S. Humphrey, Pearson Education
2. Software Project Management, Walker Royce, Pearson Education

REFERENCES:

1. An Introduction to the Team Software Process, Watts S. Humphrey, Pearson Education, 2000 Process Improvement essentials, James R. Persse, O'Reilly, 2006
2. Software Project Management, Bob Hughes & Mike Cotterell, fourth edition, TMH, 2006
3. Applied Software Project Management, Andrew Stellman & Jennifer Greene, O'Reilly, 2006.
4. Head First PMP, Jennifer Greene & Andrew Stellman, O'Reilly, 2007
5. Software Engineering Project Management, Richard H. Thayer & Edward Yourdon, 2nd edition, Wiley India, 2004.
6. Agile Project Management, Jim Highsmith, Pearson education, 2004.

**COMPUTATIONAL COMPLEXITY
(PROFESSIONAL ELECTIVE – IV)**

B.Tech. IV Year I Sem.
Course Code: CS741PE

L	T	P	C
3	0	0	3

Prerequisites:

1. A course on “Computer Programming and Data Structures”.
2. A course on “Discrete Structures and Graph Theory”.

Course Objectives:

- Introduces to theory of computational complexity classes
- Discuss about algorithmic techniques and application of these techniques to problems.
- Introduce to randomized algorithms and discuss how effective they are in reducing time and space complexity.
- Discuss about Graph based algorithms and approximation algorithms
- Discuss about search trees

Course Outcomes:

- Ability to classify decision problems into appropriate complexity classes
- Ability to specify what it means to reduce one problem to another, and construct reductions for simple examples.
- Ability to classify optimization problems into appropriate approximation complexity classes
- Ability to choose appropriate data structure for the given problem
- Ability to choose and apply appropriate design method for the given problem

UNIT - I

Computational Complexity: Polynomial time and its justification, Nontrivial examples of polynomial-time algorithms, the concept of reduction (reducibility), Class P Class NP and NP- Completeness, The P versus NP problem and why it’s hard

UNIT - II

Algorithmic paradigms: Dynamic Programming – Longest common subsequence, matrix chain multiplication, knapsack problem, Greedy – 0-1 knapsack, fractional knapsack, scheduling problem, Huffman coding, MST, Branch-and-bound – travelling sales person problem, 0/1 knapsack problem, Divide and Conquer – Merge sort, binary search, quick sort.

UNIT - III

Randomized Algorithms: Finger Printing, Pattern Matching, Graph Problems, Algebraic Methods, Probabilistic Primality Testing, De-Randomization Advanced Algorithms:

UNIT - IV

Graph Algorithms: Shortest paths, Flow networks, Spanning Trees; Approximation algorithms, Randomized algorithms. Approximation algorithms: Polynomial Time Approximation Schemes.

UNIT - V

Advanced Data Structures and applications: Decision Trees and Circuits, B-Trees, AVL Trees, Red and Black trees, Dictionaries and tries, Maps, Binomial Heaps, Fibonacci Heaps, Disjoint sets, Union by Rank and Path Compression

TEXT BOOKS:

1. T. Cormen, C. Leiserson, R. Rivest and C. Stein, Introduction to Algorithms, Third Edition, McGraw-Hill, 2009.
2. R. Motwani and P. Raghavan, Randomized Algorithms, Cambridge University Press, 1995.
3. J. J. McConnell, Analysis of Algorithms: An Active Learning Approach, Jones & Bartlett Publishers, 2001.
4. D. E. Knuth, Art of Computer Programming, Volume 3, Sorting and Searching, Second Edition, Addison-Wesley Professional, 1998.
5. S. Dasgupta, C. H. Papadimitriou and U. V. Vazirani, Algorithms, McGraw-Hill, 2008.

CLOUD COMPUTING
(PROFESSIONAL ELECTIVE – IV)

B.Tech. IV Year I Sem.
Course Code: CS742PE

L	T	P	C
3	0	0	3

Prerequisites:

- A course on “Computer Networks”.
- A course on “Operating Systems”.
- A course on “Distributed Systems”.

Course Objectives:

- This course provides an insight into cloud computing
- Topics covered include- distributed system models, different cloud service models, service-oriented architectures, cloud programming and software environments, resource management.

Course Outcomes:

- Ability to understand various service delivery models of a cloud computing architecture.
- Ability to understand the ways in which the cloud can be programmed and deployed.
- Understanding cloud service providers.

UNIT - I

Computing Paradigms: High-Performance Computing, Parallel Computing, Distributed Computing, Cluster Computing, Grid Computing, Cloud Computing, Bio computing, Mobile Computing, Quantum Computing, Optical Computing, Nano computing.

UNIT - II

Cloud Computing Fundamentals: Motivation for Cloud Computing, The Need for Cloud Computing, Defining Cloud Computing, Definition of Cloud computing, Cloud Computing Is a Service, Cloud Computing Is a Platform, Principles of Cloud computing, Five Essential Characteristics, Four Cloud Deployment Models

UNIT - III

Cloud Computing Architecture and Management: Cloud architecture, Layer, Anatomy of the Cloud, Network Connectivity in Cloud Computing, Applications, on the Cloud, Managing the Cloud, Managing the Cloud Infrastructure Managing the Cloud application, Migrating Application to Cloud, Phases of Cloud Migration Approaches for Cloud Migration.

UNIT - IV

Cloud Service Models: Infrastructure as a Service, Characteristics of IaaS. Suitability of IaaS, Pros and Cons of IaaS, Summary of IaaS Providers, Platform as a Service,

Characteristics of PaaS, Suitability of PaaS, Pros and Cons of PaaS, Summary of PaaS Providers, Software as a Service, Characteristics of SaaS, Suitability of SaaS, Pros and Cons of SaaS, Summary of SaaS Providers, Other Cloud Service Models.

UNIT - V

Cloud Service Providers: EMC, EMC IT, Captiva Cloud Toolkit, Google, Cloud Platform, Cloud Storage, Google Cloud Connect, Google Cloud Print, Google App Engine, Amazon Web Services, Amazon Elastic Compute Cloud, Amazon Simple Storage Service, Amazon Simple Queue ,service, Microsoft, Windows Azure, Microsoft Assessment and Planning Toolkit, SharePoint, IBM, Cloud Models, IBM Smart Cloud, SAP Labs, SAP HANA Cloud Platform, Virtualization Services Provided by SAP, Sales force, Sales Cloud, Service Cloud: Knowledge as a Service, Rack space, VMware, Manjra soft, Aneka Platform

TEXT BOOKS:

1. Essentials of cloud Computing: K. Chandrasekhran, CRC press, 2014

REFERENCE BOOKS:

1. Cloud Computing: Principles and Paradigms by Rajkumar Buyya, James Broberg and Andrzej M. Goscinski, Wiley, 2011.
2. Distributed and Cloud Computing, Kai Hwang, Geoffery C. Fox, Jack J. Dongarra, Elsevier, 2012.
3. Cloud Security and Privacy: An Enterprise Perspective on Risks and Compliance, Tim Mather, Subra Kumaraswamy, Shahed Latif, O'Reilly, SPD, rp2011.

**BLOCKCHAIN TECHNOLOGY
(PROFESSIONAL ELECTIVE – IV)**

B.Tech. IV Year I Sem.
Course Code: CS743PE

L T P C
3 0 0 3

Prerequisites

1. Knowledge in security and applied cryptography;
2. Knowledge in distributed databases

Course Objectives:

- To Introduce block chain technology and Cryptocurrency

Course Outcomes:

- Learn about research advances related to one of the most popular technological areas today.

UNIT- I

Introduction: Block chain or distributed trust, Protocol, Currency, Cryptocurrency, How a Cryptocurrency works, Crowdfunding

UNIT- II

Extensibility of Blockchain concepts, Digital Identity verification, Block chain Neutrality, Digital art, Blockchain Environment

UNIT- III

Blockchain Science: Gridcoin, Folding coin, Blockchain Genomics, Bitcoin MOOCs

UNIT - IV

Currency, Token, Tokenizing, Campuscoin, Coindrop as a strategy for Public adoption, Currency Multiplicity, Demurrage currency

UNIT - V

Technical challenges, Business model challenges, Scandals and Public perception, Government Regulations

TEXTBOOK:

1. Blockchain Blue print for Economy by Melanie Swan

REFERENCE:

1. Blockchain Basics: A Non-Technical Introduction in 25 Steps 1st Edition, by Daniel Drescher

**SOCIAL NETWORK ANALYSIS
(PROFESSIONAL ELECTIVE – IV)**

B.Tech. IV Year I Sem.
Course Code: CS744PE

L	T	P	C
3	0	0	3

Course Objectives:

- To understand the concept of semantic web and related applications.
- To learn knowledge representation using ontology.
- To understand human behaviour in social web and related communities.
- To learn visualization of social networks.

Course Outcomes: Upon completion of the course, the students should be able to:

- Develop semantic web related applications.
- Represent knowledge using ontology.
- Predict human behaviour in social web and related communities.
- Visualize social networks.

UNIT - I

INTRODUCTION

Introduction to Semantic Web: Limitations of current Web – Development of Semantic Web – Emergence of the Social Web – Social Network analysis: Development of Social Network Analysis – Key concepts and measures in network analysis – Electronic sources for network analysis: Electronic discussion networks, Blogs and online communities – Web-based networks – Applications of Social Network Analysis.

UNIT - II

Modelling, Aggregating and Knowledge Representation

Ontology and their role in the Semantic Web: Ontology-based knowledge Representation – Ontology languages for the Semantic Web: Resource Description Framework – Web Ontology Language – Modelling and aggregating social network data: State-of-the-art in network data representation – Ontological representation of social individuals – Ontological representation of social relationships – Aggregating and reasoning with social network data – Advanced representations.

UNIT - III

Extraction and Mining Communities in Web Social Networks

Extracting evolution of Web Community from a Series of Web Archive – Detecting communities in social networks – Definition of community – Evaluating communities – Methods for community detection and mining – Applications of community mining algorithms – Tools for detecting communities social network infrastructures and communities – Decentralized online social networks – Multi-Relational characterization of dynamic social network communities.

UNIT - IV

Predicting Human Behaviour and Privacy Issues

Understanding and predicting human behaviour for social communities – User data management – Inference and Distribution – Enabling new human experiences – Reality mining – Context – Awareness – Privacy in online social networks – Trust in online environment – Trust models based on subjective logic – Trust network analysis – Trust transitivity analysis – Combining trust and reputation – Trust derivation based on trust comparisons – Attack spectrum and countermeasures.

UNIT - V

Visualization and Applications of Social Networks

Graph theory – Centrality – Clustering – Node-Edge Diagrams – Matrix representation – Visualizing online social networks, Visualizing social networks with matrix-based representations – Matrix and Node-Link Diagrams – Hybrid representations – Applications – Cover networks – Community welfare – Collaboration networks – Co-Citation networks.

TEXT BOOKS:

1. Peter Mika, —Social Networks and the Semantic Web, First Edition, Springer 2007.
2. Borko Furht, —Handbook of Social Network Technologies and Applications, 1st Edition, Springer, 2010.

REFERENCES:

1. Guandong Xu, Yanchun Zhang and Lin Li, Web Mining and Social Networking – Techniques and applications, First Edition, Springer, 2011.
2. Dion Goh and Schubert Foo - Social information Retrieval Systems: Emerging Technologies and Applications for Searching the Web Effectively, IGI Global Snippet, 2008.
3. Max Chevalier, Christine Julien and Chantal Soulé-Dupuy, Collaborative and Social Information Retrieval and Access: Techniques for Improved user Modelling, IGI Global Snippet, 2009.
4. John G. Breslin, Alexander Passant and Stefan Decker, -The Social Semantic Web, Springer, 2009.

DATA MINING LAB

B.Tech. IV Year I Sem.
Course Code: CS703PC

L	T	P	C
0	0	3	2

Course Objectives:

- To obtain practical experience using data mining techniques on real world data sets.
- Emphasize hands-on experience working with all real data sets.

Course Outcomes:

- Ability to add mining algorithms as a component to the existing tools
- Ability to apply mining techniques for realistic data.

List of Sample Problems:**Task 1: Credit Risk Assessment****Description:**

The business of banks is making loans. Assessing the credit worthiness of an applicant is of crucial importance. You have to develop a system to help a loan officer decide whether the credit of a customer is good, or bad. A bank's business rules regarding loans must consider two opposing factors. On the one hand, a bank wants to make as many loans as possible. Interest on these loans is the banks profit source. On the other hand, a bank cannot afford to make too many bad loans. Too many bad loans could lead to the collapse of the bank. The bank's loan policy must involve a compromise: not too strict, and not too lenient. To do the assignment, you first and foremost need some knowledge about the world of credit. You can acquire such knowledge in a number of ways.

1. Knowledge Engineering. Find a loan officer who is willing to talk. Interview her and try to represent her knowledge in the form of production rules.
2. Books. Find some training manuals for loan officers or perhaps a suitable textbook on finance. Translate this knowledge from text form to production rule form.
3. Common sense. Imagine yourself as a loan officer and make up reasonable rules which can be used to judge the credit worthiness of a loan applicant.
4. Case histories. Find records of actual cases where competent loan officers correctly judged when, and when not to, approve a loan application.

The German Credit Data:

Actual historical credit data is not always easy to come by because of confidentiality rules. Here is one such dataset, consisting of 1000 actual cases collected in Germany. Credit dataset (original) Excel spreadsheet version of the German credit data. In spite of the fact that the data is German, you should probably make use of it for this assignment. (Unless you really can consult a real loan officer!)

A few notes on the German dataset

1. DM stands for Deutsche Mark, the unit of currency, worth about 90 cents Canadian (but looks and acts like a quarter).

2. owns_telephone. German phone rates are much higher than in Canada so fewer people own telephones.
3. foreign_worker. There are millions of these in Germany (many from Turkey). It is very hard to get German citizenship if you were not born of German parents.
4. There are 20 attributes used in judging a loan applicant. The goal is to classify the applicant into one of two categories, good or bad.

Subtasks: (Turn in your answers to the following tasks)

1. List all the categorical (or nominal) attributes and the real-valued attributes separately. (5 marks)
2. What attributes do you think might be crucial in making the credit assessment? Come up with some simple rules in plain English using your selected attributes. (5 marks)
3. One type of model that you can create is a Decision Tree - train a Decision Tree using the complete dataset as the training data. Report the model obtained after training. (10 marks)
4. Suppose you use your above model trained on the complete dataset, and classify credit good/bad for each of the examples in the dataset. What % of examples can you classify correctly? (This is also called testing on the training set) Why do you think you cannot get 100 % training accuracy? (10 marks)
5. Is testing on the training set as you did above a good idea? Why or Why not ? (10 marks)
6. One approach for solving the problem encountered in the previous question is using cross-validation? Describe what is cross-validation briefly. Train a Decision Tree again using cross-validation and report your results. Does your accuracy increase/decrease? Why? (10 marks)
7. Check to see if the data shows a bias against "foreign workers" (attribute 20), or "personal-status" (attribute 9). One way to do this (perhaps rather simple minded) is to remove these attributes from the dataset and see if the decision tree created in those cases is significantly different from the full dataset case which you have already done. To remove an attribute, you can use the preprocess tab in Weka's GUI Explorer. Did removing these attributes have any significant effect? Discuss. (10 marks)
8. Another question might be, do you really need to input so many attributes to get good results? Maybe only a few would do. For example, you could try just having attributes 2, 3, 5, 7, 10, 17 (and 21, the class attribute (naturally)). Try out some combinations. (You had removed two attributes in problem 7. Remember to reload the arff data file to get all the attributes initially before you start selecting the ones you want.) (10 marks)
9. Sometimes, the cost of rejecting an applicant who actually has a good credit (case 1) might be higher than accepting an applicant who has bad credit (case 2). Instead of counting the misclassifications equally in both cases, give a higher cost to the first case (say cost 5) and lower cost to the second case. You can do this by using a cost matrix in Weka. Train your Decision Tree again and report the Decision Tree and cross-validation results. Are they significantly different from results obtained in problem 6 (using equal cost)? (10 marks)

10. Do you think it is a good idea to prefer simple decision trees instead of having long complex decision trees? How does the complexity of a Decision Tree relate to the bias of the model? (10 marks)
11. You can make your Decision Trees simpler by pruning the nodes. One approach is to use Reduced Error Pruning - Explain this idea briefly. Try reduced error pruning for training your Decision Trees using cross-validation (you can do this in Weka) and report the Decision Tree you obtain? Also, report your accuracy using the pruned model. Does your accuracy increase? (10 marks)
- 12.(Extra Credit): How can you convert a Decision Trees into "if-then-else rules". Make up your own small Decision Tree consisting of 2-3 levels and convert it into a set of rules. There also exist different classifiers that output the model in the form of rules - one such classifier in Weka is rules. PART, train this model and report the set of rules obtained. Sometimes just one attribute can be good enough in making the decision, yes, just one ! Can you predict what attribute that might be in this dataset ? OneR classifier uses a single attribute to make decisions (it chooses the attribute based on minimum error). Report the rule obtained by training a one R classifier. Rank the performance of j48, PART and oneR. (10 marks)

Task Resources:

- Mentor lecture on Decision Trees
- Andrew Moore's Data Mining Tutorials (See tutorials on Decision Trees and Cross Validation)
- Decision Trees (Source: Tan, MSU)
- Tom Mitchell's book slides (See slides on Concept Learning and Decision Trees)
- Weka resources:
 - Introduction to Weka (html version) (download ppt version)
 - Download Weka
 - Weka Tutorial
 - ARFF format
 - Using Weka from command line

Task 2: Hospital Management System

Data Warehouse consists Dimension Table and Fact Table.

REMEMBER The following

Dimension

The dimension object (Dimension):

_ Name

_ Attributes (Levels) , with one primary key

_ Hierarchies

One time dimension is must.

About Levels and Hierarchies

Dimension objects (dimension) consist of a set of levels and a set of hierarchies defined over those levels. The levels represent levels of aggregation. Hierarchies describe parent-child relationships among a set of levels.

For example, a typical calendar dimension could contain five levels. Two hierarchies can be defined on these levels:

H1: YearL > QuarterL > MonthL > WeekL > DayL

H2: YearL > WeekL > DayL

The hierarchies are described from parent to child, so that Year is the parent of Quarter, Quarter the parent of Month, and so forth.

About Unique Key Constraints

When you create a definition for a hierarchy, Warehouse Builder creates an identifier key for each level of the hierarchy and a unique key constraint on the lowest level (Base Level)

Design a Hospital Management system data warehouse (TARGET) consists of Dimensions Patient, Medicine, Supplier, Time. Where measures are 'NO UNITS', UNIT PRICE.

Assume the Relational database (SOURCE) table schemas as follows

TIME (day, month, year),

PATIENT (patient_name, Age, Address, etc.,)

MEDICINE (Medicine_Brand_name, Drug_name, Supplier, no_units, Uinit_Price, etc.,)

SUPPLIER :(Supplier_name, Medicine_Brand_name, Address, etc.,)

If each Dimension has 6 levels, decide the levels and hierarchies, Assume the level names suitably.

Design the Hospital Management system data warehouse using all schemas. Give the example 4-D cube with assumption names.

PYTHON PROGRAMMING LAB

B.Tech. IV Year I Sem.
Course Code: CS751PC

L	T	P	C
0	0	3	2

Prerequisites: Students should install Python on Linux platform.

Course Objectives:

- To be able to introduce core programming basics and program design with functions using Python programming language.
- To understand a range of Object-Oriented Programming, as well as in-depth data and information processing techniques.
- To understand the high-performance programs designed to strengthen the practical expertise.

Course Outcomes:

- Student should be able to understand the basic concepts scripting and the contributions of scripting language
- Ability to explore python especially the object oriented concepts, and the built in objects of Python.
- Ability to create practical and contemporary applications such as TCP/IP network programming, Web applications, discrete event simulations

List of Programs:

1. Write a program to demonstrate different number data types in Python.
2. Write a program to perform different Arithmetic Operations on numbers in Python.
3. Write a program to create, concatenate and print a string and accessing sub-string from a given string.
4. Write a python script to print the current date in the following format “Sun May 29 02:26:23 IST 2017”
5. Write a program to create, append, and remove lists in python.
6. Write a program to demonstrate working with tuples in python.
7. Write a program to demonstrate working with dictionaries in python.
8. Write a python program to find largest of three numbers.
9. Write a Python program to convert temperatures to and from Celsius, Fahrenheit.
[Formula: $c/5 = f-32/9$]
10. Write a Python program to construct the following pattern, using a nested for loop

```

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

```


* *

*

11. Write a Python script that prints prime numbers less than 20.
12. Write a python program to find factorial of a number using Recursion.
13. Write a program that accepts the lengths of three sides of a triangle as inputs. The program output should indicate whether or not the triangle is a right triangle (Recall from the Pythagorean Theorem that in a right triangle, the square of one side equals the sum of the squares of the other two sides).
14. Write a python program to define a module to find Fibonacci Numbers and import the module to another program.
15. Write a python program to define a module and import a specific function in that module to another program.
16. Write a script named **copyfile.py**. This script should prompt the user for the names of two text files. The contents of the first file should be input and written to the second file.
17. Write a program that inputs a text file. The program should print all of the unique words in the file in alphabetical order.
18. Write a Python class to convert an integer to a roman numeral.
19. Write a Python class to implement $\text{pow}(x, n)$
20. Write a Python class to reverse a string word by word.

MOBILE APPLICATION DEVELOPMENT LAB

B.Tech. IV Year I Sem.
Course Code: CS752PC

L T P C
0 0 3 2

Course Objectives:

- To learn how to develop Applications in android environment.
- To learn how to develop user interface applications.
- To learn how to develop URL related applications.

The student is expected to be able to do the following problems, though not limited.

1. (a) Create an Android application that shows Hello + name of the user and run it on an emulator.
 (b) Create an application that takes the name from a text box and shows hello message along with the name entered in text box, when the user clicks the OK button.
2. Create a screen that has input boxes for User Name, Password, Address, Gender (radio buttons for male and female), Age (numeric), Date of Birth (Date Picket), State (Spinner) and a Submit button. On clicking the submit button, print all the data below the Submit Button. Use
 - (a) Linear Layout, (b) Relative Layout and
 - (c) Grid Layout or Table Layout.
3. Develop an application that shows names as a list and on selecting a name it should show the details of the candidate on the next screen with a “Back” button. If the screen is rotated to landscape mode (width greater than height), then the screen should show list on left fragment and details on right fragment instead of second screen with back button. Use Fragment transactions and Rotation event listener.
4. Develop an application that uses a menu with 3 options for dialing a number, opening a website and to send an SMS. On selecting an option, the appropriate action should be invoked using intents.
5. Develop an application that inserts some notifications into Notification area and whenever a notification is inserted, it should show a toast with details of the notification.
6. Create an application that uses a text file to store user names and passwords (tab separated fields and one record per line). When the user submits a login name and password through a screen, the details should be verified with the text file data and if they match, show a dialog saying that login is successful. Otherwise, show the dialog with Login Failed message.

7. Create a user registration application that stores the user details in a database table.
8. Create a database and a user table where the details of login names and passwords are stored. Insert some names and passwords initially. Now the login details entered by the user should be verified with the database and an appropriate dialog should be shown to the user.

Note:

Android Application Development with MIT App Inventor: For the first one week, the student is advised to go through the App Inventor from MIT which gives insight into the various properties of each component.

The student should pay attention to the properties of each component, which are used later in Android programming. Following are useful links:

1. <http://ai2.appinventor.mit.edu>
2. https://drive.google.com/file/d/0B8rTtW_91YcITWF4czdBMEpZcWs/view

WEB SCRIPTING LANGUAGES LAB

B.Tech. IV Year I Sem.
Course Code: CS753PC

L	T	P	C
0	0	3	2

Prerequisites: Any High-level programming language (C, C++)

Course Objectives

- To Understand the concepts of scripting languages for developing web-based projects
- To understand the applications the of Ruby, TCL, Perl scripting languages

Course Outcomes

- Ability to understand the differences between Scripting languages and programming languages
- Able to gain some fluency programming in Ruby, Perl, TCL

List of Experiments

1. Write a Ruby script to create a new string which is n copies of a given string where n is a non-negative integer
2. Write a Ruby script which accept the radius of a circle from the user and compute the parameter and area.
3. Write a Ruby script which accept the user's first and last name and print them in reverse order with a space between them
4. Write a Ruby script to accept a filename from the user print the extension of that
5. Write a Ruby script to find the greatest of three numbers
6. Write a Ruby script to print odd numbers from 10 to 1
7. Write a Ruby script to check two integers and return true if one of them is 20 otherwise return their sum
8. Write a Ruby script to check two temperatures and return true if one is less than 0 and the other is greater than 100
9. Write a Ruby script to print the elements of a given array
10. Write a Ruby program to retrieve the total marks where subject name and marks of a student stored in a hash
11. Write a TCL script to find the factorial of a number
12. Write a TCL script that multiplies the numbers from 1 to 10
13. Write a TCL script for Sorting a list using a comparison function
14. Write a TCL script to (i)create a list (ii)append elements to the list (iii)Traverse the list (iv)Concatenate the list
15. Write a TCL script to comparing the file modified times.
16. Write a TCL script to Copy a file and translate to native format.
17. a) Write a Perl script to find the largest number among three numbers.
b) Write a Perl script to print the multiplication tables from 1-10 using subroutines.
18. Write a Perl program to implement the following list of manipulating functions
a) Shift

b) Unshift

c) Push

19. a) Write a Perl script to substitute a word, with another word in a string.

b) Write a Perl script to validate IP address and email address.

20. Write a Perl script to print the file in reverse order using command line arguments

INTERNET OF THINGS LAB

B.Tech. IV Year I Sem.
Course Code: CS754PC

L T P C
0 0 3 2

Following are some of the programs that a student should be able to write and test on an Raspberry Pi, but not limited to this only.

- 1 Start Raspberry Pi and try various Linux commands in command terminal window:
ls, cd, touch, mv, rm, man, mkdir, rmdir, tar, gzip, cat, more, less, ps, sudo, cron, chown, chgrp, ping etc.
2. Run some python programs on Pi like:
 - Read your name and print Hello message with name
 - Read two numbers and print their sum, difference, product and division.
 - Word and character count of a given string
 - Area of a given shape (rectangle, triangle and circle) reading shape and appropriate values from standard input
 - Print a name 'n' times, where name and n are read from standard input, using for and while loops.
 - Handle Divided by Zero Exception.
 - Print current time for 10 times with an interval of 10 seconds.
 - Read a file line by line and print the word count of each line.
3. Light an LED through Python program
4. Get input from two switches and switch on corresponding LEDs
5. Flash an LED at a given on time and off time cycle, where the two times are taken from a file.
6. Flash an LED based on cron output (acts as an alarm)
7. Switch on a relay at a given time using cron, where the relay's contact terminals are connected to a load.
8. Get the status of a bulb at a remote place (on the LAN) through web.

The student should have hands on experience in using various sensors like temperature, humidity, smoke, light, etc. and should be able to use control web camera, network, and relays connected to the Pi.

INFORMATION THEORY AND CODING
(PROFESSIONAL ELECTIVE – V)

B.Tech. IV Year II Sem.
Course Code: CS851PE

L T P C
3 0 0 3

Prerequisite: Digital Communications

Course Objectives:

- To acquire the knowledge in measurement of information and errors.
- Understand the importance of various codes for communication systems
- To design encoder and decoder of various codes.
- To know the applicability of source and channel codes

Course Outcomes: Upon completing this course, the student will be able to

- Learn measurement of information and errors.
- Obtain knowledge in designing various source codes and channel codes
- Design encoders and decoders for block and cyclic codes
- Understand the significance of codes in various applications

UNIT - I

Coding for Reliable Digital Transmission and storage

Mathematical model of Information, A Logarithmic Measure of Information, Average and Mutual Information and Entropy, Types of Errors, Error Control Strategies.

Source Codes: Shannon-fano coding, Huffman coding

UNIT - II

Linear Block Codes: Introduction to Linear Block Codes, Syndrome and Error Detection, Minimum Distance of a Block code, Error-Detecting and Error-correcting Capabilities of a Block code, Standard array and Syndrome Decoding, Probability of an undetected error for Linear Codes over a BSC, Hamming Codes. Applications of Block codes for Error control in data storage system

UNIT - III

Cyclic Codes: Description, Generator and Parity-check Matrices, Encoding, Syndrome Computation and Error Detection, Decoding, Cyclic Hamming Codes, shortened cyclic codes, Error-trapping decoding for cyclic codes, Majority logic decoding for cyclic codes.

UNIT - IV

Convolutional Codes: Encoding of Convolutional Codes- Structural and Distance Properties, state, tree, trellis diagrams, maximum likelihood decoding, Sequential decoding, Majority- logic decoding of Convolution codes. Application of Viterbi Decoding and Sequential Decoding, Applications of Convolutional codes in ARQ system.

UNIT - V

BCH Codes: Minimum distance and BCH bounds, Decoding procedure for BCH codes, Syndrome computation and iterative algorithms, Error locations polynomials for single and double error correction.

TEXT BOOKS:

1. Error Control Coding- Fundamentals and Applications –Shu Lin, Daniel J. Costello, Jr, Prentice Hall, Inc 2014.
2. Error Correcting Coding Theory-Man Young Rhee, McGraw – Hill Publishing 1989

REFERENCE BOOKS:

1. Digital Communications- John G. Proakis, 5th ed, TMH 2008.
2. Introduction to Error Control Codes-Salvatore Gravano-oxford
3. Error Correction Coding – Mathematical Methods and Algorithms – Todd K. Moon, 2006, Wiley India.
4. Information Theory, Coding and Cryptography – Ranjan Bose, 2nd Edition, 2009, TMH.

REAL-TIME SYSTEMS
(PROFESSIONAL ELECTIVE – V)

B.Tech. IV Year II Sem.
Course Code: CS852PE

L T P C
3 0 0 3

Prerequisite: Computer Organization and Operating System

Course Objectives:

- To provide broad understanding of the requirements of Real Time Operating Systems.
- To make the student understand, applications of these Real Time features using case studies.

Course Outcomes:

- Be able to explain real-time concepts such as preemptive multitasking, task priorities, priority inversions, mutual exclusion, context switching, and synchronization, interrupt
- latency and response time, and semaphores.
- Able describe how a real-time operating system kernel is implemented.
- Able explain how tasks are managed.
- Explain how the real-time operating system implements time management.
- Discuss how tasks can communicate using semaphores, mailboxes, and queues.
- Be able to implement a real-time system on an embedded processor.
- Be able to work with real time operating systems like RT Linux, Vx Works, MicroC /OSII, Tiny OS

UNIT – I: Introduction

Introduction to UNIX/LINUX, Overview of Commands, File I/O, (open, create, close, lseek, read, write), Process Control (fork, vfork, exit, wait, waitpid, exec).

UNIT - II: Real Time Operating Systems

Brief History of OS, Defining RTOS, The Scheduler, Objects, Services, Characteristics of RTOS, defining a Task, asks States and Scheduling, Task Operations, Structure, Synchronization, Communication and Concurrency. Defining Semaphores, Operations and Use, Defining Message Queue, States, Content, Storage, Operations and Use

UNIT - III: Objects, Services and I/O

Pipes, Event Registers, Signals, Other Building Blocks, Component Configuration, Basic I/O Concepts, I/O Subsystem

UNIT - IV: Exceptions, Interrupts and Timers

Exceptions, Interrupts, Applications, Processing of Exceptions and Spurious Interrupts, Real Time Clocks, Programmable Timers, Timer Interrupt Service Routines (ISR), Soft Timers,

Operations.

UNIT - V: Case Studies of RTOS

RT Linux, MicroC/OS-II, Vx Works, Embedded Linux, and Tiny OS.

TEXT BOOKS:

1. Real Time Concepts for Embedded Systems – Qing Li, Elsevier, 2011

REFERENCE BOOKS:

1. Embedded Systems- Architecture, Programming and Design by Rajkamal, 2007, TMH.
2. Advanced UNIX Programming, Richard Stevens
3. Embedded Linux: Hardware, Software and Interfacing – Dr. Craig Hollabaugh

DATA ANALYTICS
(PROFESSIONAL ELECTIVE – V)

B.Tech. IV Year II Sem.
Course Code: CS853PE

L	T	P	C
3	0	0	3

Prerequisites

1. A course on “Database Management Systems”
2. Knowledge of probability and statistics

Course Objectives: To explore the fundamental concepts of data analytics.

- To learn the principles and methods of statistical analysis
- Discover interesting patterns, analyze supervised and unsupervised models and estimate the accuracy of the algorithms.
- To understand the various search methods and visualization techniques.

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

- Understand the impact of data analytics for business decisions and strategy
- Carry out data analysis/statistical analysis
- To carry out standard data visualization and formal inference procedures
- Design Data Architecture
- Understand various Data Sources

UNIT - I

Data Management: Design Data Architecture and manage the data for analysis, understand various sources of Data like Sensors/Signals/GPS etc. Data Management, Data Quality (noise, outliers, missing values, duplicate data) and Data Processing & Processing.

UNIT - II

Data Analytics: Introduction to Analytics, Introduction to Tools and Environment, Application of Modeling in Business, Databases & Types of Data and variables, Data Modeling Techniques, Missing Imputations etc. Need for Business Modeling.

UNIT - III

Regression – Concepts, Blue property assumptions, Least Square Estimation, Variable Rationalization, and Model Building etc.

Logistic Regression: Model Theory, Model fit Statistics, Model Construction, Analytics applications to various Business Domains etc.

UNIT - IV

Object Segmentation: Regression Vs Segmentation – Supervised and Unsupervised Learning, Tree Building – Regression, Classification, Overfitting, Pruning and Complexity, Multiple Decision Trees etc.

Time Series Methods: Arima, Measures of Forecast Accuracy, STL approach, Extract features from generated model as Height, Average Energy etc and Analyze for prediction

UNIT - V

Data Visualization: Pixel-Oriented Visualization Techniques, Geometric Projection Visualization Techniques, Icon-Based Visualization Techniques, Hierarchical Visualization Techniques, Visualizing Complex Data and Relations.

TEXT BOOKS:

1. Student's Handbook for Associate Analytics – II, III.
2. Data Mining Concepts and Techniques, Han, Kamber, 3rd Edition, Morgan Kaufmann Publishers.

REFERENCE BOOKS:

1. Introduction to Data Mining, Tan, Steinbach and Kumar, Addison Wesley, 2006.
2. Data Mining Analysis and Concepts, M. Zaki and W. Meira
3. Mining of Massive Datasets, Jure Leskovec Stanford Univ. Anand Rajaraman Millway Labs Jeffrey D Ullman Stanford Univ.

**MODERN SOFTWARE ENGINEERING
(PROFESSIONAL ELECTIVE – V)**

B.Tech. IV Year II Sem.
Course Code: CS854PE

L	T	P	C
3	0	0	3

UNIT - I**Introduction Extreme Programming (XP) - Agile Development**

Why Agile - Understanding Success, Beyond Deadlines, Importance of Organizational Success, Introduction to Agility, How to Be Agile - Agile methods, Don't make your own method, Road to mastery, Understanding XP (Extreme Programming) - XP life cycle, XP team, XP Concepts, Adopting XP - Knowing whether XP is suitable, Implementing XP, assessing Agility, Practicing XP - Thinking - Pair Programming, Energized work, Informative Workspace, Root cause Analysis, Retrospectives

UNIT - II

Collaborating: Trust, Sit together, Real customer involvement, Ubiquitous language, meetings, coding standards, Iteration demo, Reporting

UNIT - III

Releasing: Bugfree Release, Version Control, fast build, continuous integration, Collective ownership, Documentation

UNIT - IV

Planning: Version, Release Plan, Risk Management, Iteration Planning, Slack, Stories, Estimating

UNIT - V

Developing: Incremental requirements, Customer tests, Test driven development, Refactoring, Incremental design and architecture, spike solutions, Performance optimization, Exploratory testing

TEXT BOOK:

1. The art of Agile Development, James Shore and Shane Warden, 11th Indian Reprint, O'Reilly, 2018

REFERENCES:

1. Learning Agile, Andrew Stellman and Jennifer Greene, O'Reilly, 4th Indian Reprint, 2018
2. Practices of an Agile Developer, Venkat Subramaniam and Andy Hunt, SPD, 5th Indian Reprint, 2015
3. Agile Project Management - Jim Highsmith, Pearson Low price Edition 2004

**ADVANCED ALGORITHMS
(PROFESSIONAL ELECTIVE – VI)**

B.Tech. IV Year II Sem.
Course Code: CS861PE

L	T	P	C
3	0	0	3

Prerequisites:

1. A course on “Computer Programming & Data Structures”.
2. A course on “Advanced Data Structures & Algorithms”.

Course Objectives:

- Introduces the recurrence relations for analyzing the algorithms.
- Introduces the graphs and their traversals.
- Describes major algorithmic techniques (divide-and-conquer, greedy, dynamic programming, Brute Force, Transform and Conquer approaches) and mention problems for which each technique is appropriate.
- Describes how to evaluate and compare different algorithms using worst-case, average-case and best-case analysis.
- Introduces string matching algorithms.
- Introduces linear programming.

Course Outcomes:

- Ability to analyze the performance of algorithms.
- Ability to choose appropriate data structures and algorithm design methods for a specified application.
- Ability to understand how the choice of data structures and the algorithm design methods impact the performance of programs.

UNIT - I

Introduction: Role of Algorithms in computing, Order Notation, Recurrences, Probabilistic Analysis and Randomized Algorithms. Sorting and Order Statistics: Heap sort, Quick sort and Sorting in Linear Time.

Advanced Design and Analysis Techniques: Dynamic Programming- Matrix chain Multiplication, Longest common Subsequence and optimal binary Search trees.

UNIT - II

Greedy Algorithms - Huffman Codes, Activity Selection Problem. Amortized Analysis.

Graph Algorithms: Topological Sorting, Minimum Spanning trees, Single Source Shortest Paths, Maximum Flow algorithms.

UNIT - III

Sorting Networks: Comparison Networks, Zero-one principle, bitonic Sorting Networks, Merging Network, Sorting Network.

Matrix Operations- Strassen's Matrix Multiplication, inverting matrices, Solving system of linear Equations

UNIT - IV

String Matching: Naive String Matching, Rabin-Karp algorithm, matching with finite Automata, Knuth- Morris - Pratt algorithm.

UNIT - V

NP-Completeness and Approximation Algorithms: Polynomial time, polynomial time verification, NP-Completeness and reducibility, NP-Complete problems. Approximation Algorithms- Vertex cover Problem, Travelling Sales person problem

TEXT BOOK:

1. Introduction to Algorithms," T. H. Cormen, C.E. Leiserson, R. L. Rivest, and C. Stein, Third Edition, PHI.

REFERENCES:

1. Fundamentals of Computer Algorithms, Ellis Horowitz, Satraj Sahni and Rajasekharam, Galgotia publications pvt. Ltd.
2. Design and Analysis Algorithms - Parag Himanshu Dave, Himanshu Bhalchandra Dave Publisher: Pearson
3. Algorithm Design: Foundations, Analysis and Internet examples, M.T. Goodrich and R. Tomassia, John Wiley and sons.
4. Data structures and Algorithm Analysis in C++, Allen Weiss, Second edition, Pearson education.

**WEB SERVICES AND SERVICE ORIENTED ARCHITECTURE
(PROFESSIONAL ELECTIVE – VI)**

B.Tech. IV Year II Sem.
Course Code: CS862PE

L	T	P	C
3	0	0	3

Course Objectives:

- To Understand Web Services and implementation model for SOA
- To Understand the SOA, its Principles and Benefits
- To Understand XML concepts
- To Understand paradigms needed for testing Web Services
- To explore different Test Strategies for SOA-based applications
- To implement functional testing, compliance testing and load testing of Web Services
- To Identify bug-finding ideas in testing Web Services

UNIT - I

Evolution and Emergence of Web Services - Evolution of distributed computing, Core distributed computing technologies – client/server, CORBA, JAVA RMI, Microsoft DCOM, MOM, Challenges in Distributed Computing, role of J2EE and XML in distributed computing, emergence of Web Services and Service Oriented Architecture (SOA).

Introduction to Web Services – The definition of web services, basic operational model of web services, tools and technologies enabling web services, benefits and challenges of using web services.

UNIT - II

Web Services Architecture – Web services Architecture and its characteristics, core building blocks of web services, standards and technologies available for implementing web services, web services communication, basic steps of implementing web services. Describing Web Services – WSDL introduction, nonfunctional service description, WSDL1.1 Vs WSDL 2.0, WSDL document, WSDL elements, WSDL binding, WSDL tools, WSDL port type, limitations of WSDL.

UNIT – III

Brief Over View of XML – XML Document structure, XML namespaces, Defining structure in XML documents, Reuse of XML schemes, Document navigation and transformation. SOAP: Simple Object Access Protocol, Inter-application communication and wire protocols, SOAP as a messaging protocol, Structure of a SOAP message, SOAP envelope, Encoding, Service Oriented Architectures, SOA revisited, Service roles in a SOA, Reliable messaging, The enterprise Service Bus, SOA Development Lifecycle, SOAP HTTP binding, SOAP communication model, Error handling in SOAP.

UNIT – IV

Registering and Discovering Services: The role of service registries, Service discovery, Universal Description, Discovery, and Integration, UDDI Architecture, UDDI Data Model,

Interfaces, UDDI Implementation, UDDI with WSDL, UDDI specification, Service Addressing and Notification, Referencing and addressing Web Services, Web Services Notification.

UNIT – V

SOA and web services security considerations, Network-level security mechanisms, Application-level security topologies, XML security standards, Semantics and Web Services, The semantic interoperability problem, The role of metadata, Service metadata, Overview of .NET and J2EE, SOA and Web Service Management, Managing Distributed System, Enterprise management Framework, Standard distributed management frameworks, Web service management, Richer schema languages, WS-Metadata Exchange.

TEXT BOOKS:

1. Web Services & SOA Principles and Technology, Second Edition, Michael P. Papazoglou.
2. Developing Java Web Services, R. Nagappan, R. Skoczylas, R.P. Sriganesh, Wiley India.
3. Developing Enterprise Web Services, S. Chatterjee, J. Webber, Pearson Education.

REFERENCE BOOKS:

1. XML, Web Services, and the Data Revolution, F.P. Coyle, Pearson Education.
2. Building web Services with Java, 2nd Edition, S. Graham and others, Pearson Education.
3. Java Web Services, D.A. Chappell & T. Jewell, O'Reilly, SPD.
4. McGovern, et al., "Java web Services Architecture", Morgan Kaufmann Publishers, 2005.
5. J2EE Web Services, Richard Monson-Haefel, Pearson Education.

COMPUTER FORENSICS
(PROFESSIONAL ELECTIVE – VI)

B.Tech. IV Year II Sem.
Course Code: CS863PE

L T P C
3 0 0 3

Prerequisite: Network Security

Course Objectives

- A brief explanation of the objective is to provide digital evidences which are obtained from digital media.
- In order to understand the objectives of computer forensics, first of all, people have to recognize the different roles computer plays in a certain crime.
- According to a snippet from the United States Security Service, the functions computer has in different kinds of crimes.

Course Outcomes

- Students will understand the usage of computers in forensic, and how to use various forensic tools for a wide variety of investigations.
- It gives an opportunity to students to continue their zeal in research in computer forensics

UNIT- I

Introduction of Cybercrime: Types, The Internet spawns crime, Worms versus viruses, Computers' roles in crimes, Introduction to digital forensics, Introduction to Incident - Incident Response Methodology –Steps - Activities in Initial Response, Phase after detection of an incident

UNIT-II

Initial Response and forensic duplication, Initial Response & Volatile Data Collection from Windows system -Initial Response & Volatile Data Collection from Unix system – Forensic Duplication: Forensic duplication: Forensic Duplicates as Admissible Evidence, Forensic Duplication Tool Requirements, Creating a Forensic. Duplicate/Qualified Forensic Duplicate of a Hard Drive

UNIT – III

Forensics analysis and validation: Determining what data to collect and analyze, validating forensic data, addressing data-hiding techniques, performing remote acquisitions

Network Forensics: Network forensics overview, performing live acquisitions, developing standard procedures for network forensics, using network tools, examining the honeynet project.

UNIT - IV

Current Forensic tools: evaluating computer forensic tool needs, computer forensics software tools, computer forensics hardware tools, validating and testing forensics software
E-Mail Investigations: Exploring the role of e-mail in investigation, exploring the roles of the client and server in e-mail, investigating e-mail crimes and violations, understanding e-mail servers, using specialized e-mail forensic tools.

Cell phone and mobile device forensics: Understanding mobile device forensics, understanding acquisition procedures for cell phones and mobile devices.

UNIT - V

Working with Windows and DOS Systems: understanding file systems, exploring Microsoft File Structures, Examining NTFS disks, Understanding whole disk encryption, windows registry, Microsoft startup tasks, MS-DOS startup tasks, virtual machines.

TEXT BOOKS:

1. Kevin Mandia, Chris Prorise, "Incident Response and computer forensics", Tata McGraw Hill, 2006.
2. Computer Forensics, Computer Crime Investigation by John R. Vacca, Firewall Media, New Delhi.
3. Computer Forensics and Investigations by Nelson, Phillips Enfinger, Steuart, CENGAGE Learning

REFERENCES:

1. Real Digital Forensics by Keith J. Jones, Richard Bejtich, Curtis W. Rose, Addison-Wesley Pearson Education
2. Forensic Compiling, A Tractitioneris Guide by Tony Sammes and Brian Jenkinson, Springer International edition.

**NEURAL NETWORKS & DEEP LEARNING
(PROFESSIONAL ELECTIVE – VI)**

B.Tech. IV Year II Sem.
Course Code: CS864PE

L	T	P	C
3	0	0	3

Course Objectives:

- To introduce the foundations of Artificial Neural Networks
- To acquire the knowledge on Deep Learning Concepts
- To learn various types of Artificial Neural Networks
- To gain knowledge to apply optimization strategies

Course Outcomes:

- Ability to understand the concepts of Neural Networks
- Ability to select the Learning Networks in modeling real world systems
- Ability to use an efficient algorithm for Deep Models
- Ability to apply optimization strategies for large scale applications

UNIT - I

Artificial Neural Networks Introduction, Basic models of ANN, important terminologies, Supervised Learning Networks, Perceptron Networks, Adaptive Linear Neuron, Back-propagation Network. Associative Memory Networks. Training Algorithms for pattern association, BAM and Hopfield Networks.

UNIT - II

Unsupervised Learning Network- Introduction, Fixed Weight Competitive Nets, Maxnet, Hamming Network, Kohonen Self-Organizing Feature Maps, Learning Vector Quantization, Counter Propagation Networks, Adaptive Resonance Theory Networks. Special Networks- Introduction to various networks.

UNIT - III

Introduction to Deep Learning, Historical Trends in Deep learning, Deep Feed - forward networks, Gradient-Based learning, Hidden Units, Architecture Design, Back-Propagation and Other Differentiation Algorithms

UNIT - IV

Regularization for Deep Learning

Parameter norm Penalties, Norm Penalties as Constrained Optimization, Regularization and Under-Constrained Problems, Dataset Augmentation, Noise Robustness, Semi-Supervised learning, Multi-task learning, Early Stopping, Parameter Typing and Parameter Sharing, Sparse Representations, Bagging and other Ensemble Methods, Dropout, Adversarial Training, Tangent Distance, tangent Prop and Manifold, Tangent Classifier

UNIT - V

Optimization for Train Deep Models

Challenges in Neural Network Optimization, Basic Algorithms, Parameter Initialization Strategies, Algorithms with Adaptive Learning Rates, Approximate Second-Order Methods, Optimization Strategies and Meta-Algorithms

Applications: Large-Scale Deep Learning, Computer Vision, Speech Recognition, Natural Language Processing

TEXT BOOKS:

1. Deep Learning: An MIT Press Book By Ian Goodfellow and Yoshua Bengio and Aaron Courville
2. Neural Networks and Learning Machines, Simon Haykin, 3rd Edition, Pearson Prentice Hall.