Autonomous Programme Structure (Modified) of
F. Y. B. Tech. (Common to All Programmes)
A. Y.: 2019-2020

F. Y. B. Tech. Semester –I

Course Code	Course Title	Teaching Scheme Hours / Week			Examination Scheme			Marks	Credit
		Lecture	Tutorial	Practical	In Semester	End Semester	Practical / Oral		
BS 1101	Engg. Mathematics - I	3	1	0	50	50	0	100	4
BS 1102	Physics - I	2	1	0	50	50	0	100	3
BS 1103	Chemistry- I	2	1	0	50	50	0	100	3
ES 1101	Basic Electrical and Electronics Engg. - I	3	0	0	50	50	0	100	3
ES 1102	Fundamentals of Programming Language - I	1	0	0	25	0	0	25	1
ES 1103	Engg. Graphics	2	0	0	25	25	0	50	2
ES 1104	Environmental Studies	2	1	0	50	50	0	100	3
BS 1104	Physics and Chemistry Lab - I	0	0	2	25	0	0	25	1
ES 1105	Basic Electrical and Electronics Engg. Lab - I	0	0	2	0	0	25	25	1
ES 1106	Fundamentals of Programming Language - I	0	0	2	0	0	25	25	1
ES 1107	Engg. Graphics Lab	0	0	2	0	0	25	25	1
NC 1101	Value Education	1	0	0	0	0	0	0	0
	Total	**16**	**4**	**8**	**325**	**275**	**75**	**675**	**23**
	Grand Total	**28**			**675**			**675**	**23**

MKSSS's Cummins College of Engineer, Karvenagar, Pune-411052

DEAN ACADEMICS
MKSSS's Cummins College of Engineering for Women Karvenagar, Pune-411052

Principal
MKSSS's Cummins College of Engg. For Women, Karvenagar, Pune-52.

APPROVED BY
Governing Body Members
MKSSS's Cummins College of Engineering for Women Karvenagar, Pune-411052

BS1101 ENGINEERING MATHEMATICS - I

Teaching Scheme:	Examination Scheme:
Lectures: 3 Hrs/Week	In-Semester : 50 Marks
Tutorial: 1 Hr/Week	End-Semester: 50 Marks
Credits: 4	

Course Objectives:

Mathematics is a necessary path to scientific knowledge which opens new perspective of mental activity. Our aim is to provide sound knowledge of engineering mathematics to make the students think mathematically and strengthen their thinking power to analyze and solve engineering problems in their respective areas.

Course Outcomes:

1. Solve the system of Linear equations by using the matrix method and apply it to check Linear Dependence, Independence of the vectors.
2. Calculate eigen values, eigen vectors and apply it to diagonalize a matrix.
3. Analyze roots of algebraic equations by applying De Moivre's theorem and analyze the function of complex numbers .
4. Compute power series expansions by using higher order derivatives.
5. Calculate partial derivatives and use to analyze maxima, minima of a given function.

Unit – I: Matrices **(07)**

Matrices, Rank of the matrix, Echelon Form, Normal form, Inverse of the matrix,System of Linear Equations, Linear Dependence and Independence, Linear

Transformations, Rotation and Translation Matrices.

Unit – II: Applications of matrices **(06)**

Eigen Values, Eigen Vectors , Cayley Hamilton Theorem , Diagonalization and applications in finding powers of matrix.

Unit–III: Complex numbers and its applications **(08)**

Argand diagrams, De moivre's theorem and its applications, Hyperbolic Functions, Separation of real and imaginary parts of functions of complex numbers, Inverse Hyperbolic Functions, Logarithm of Complex Numbers.

Unit – IV: Differential calculus **(05)**

Successive Differentiation, Method of finding nth order derivative of functions, Leibnitz theorem, Taylor's series, Maclaurin's Series.

Unit – V: Partial Differentiation **(07)**

Partial Differentiation, chain rule, composite functions, Euler's theorem on homogeneous functions,Total derivatives .

Unit – VI: Jacobian and its applications **(08)**

Jacobian, Chain rule, Partial derivatives using Jacobian, Errors and Approximations, Maxima and Minima of functions of two variables, Lagrange's method of undetermined multipliers.

Text Books:

1. B.S. Grewal, **'Higher engineering Mathematics'**, *Khanna publishers, Delhi(*40th edition),(2008).
2. B. V. Ramana, **'Higher Engineering Mathematics ',** *Tata McGraw Hill Publications* (2007)
3. Erwin Kreyszig ,**'Advanced Engineering Mathematics'** *Wiley Eastern Ltd.*(8th Student Edition)(2004).

Reference Books:

1. C.R.Wylie, L.C. Barrette, **'Advanced Engineering Mathematics',** *McGraw Hill Publi-*

*cations, New Delhi.(*6th edition)(2003)
2. Peter V. O'neil,'**Advanced Engineering Mathematics'** ,Thomson Brooks / Cole, *Singapore (*5th edition) (2007).

BS1102 PHYSICS – I

Teaching Scheme

Lectures: 2 Hrs per week

Tutorial: 1 Hr per week

Credits: 3

Examination Scheme:

In-Semester : 50 Marks

End-Semester: 50 Marks

Course Objective:

1. To introduce undergraduate students Of engineering to the principles, notions, basic physical ideas, mathematical relations and applications of Classical Physics, specifically pertaining to the theories of Electromagnetic Radiation, Optics, Special Relativity
2. To point out some of the contexts in which Classical Physics fails to account for certain experimental observation thereby requiring Quantum Physics to take over

Course Outcomes:

By taking this course, the learner will be able to –

1: **Use** the laws of Electrostatics and Electromagnetic Radiation to determine the electric field dueto static and dynamic charge distributions.
2: Apply the laws of physical optics in situations involving interference, diffraction and polarization patterns.
3: Justify the use of the principles of special relativity in situations involving elementary particles.
4: Judge the relevance of quantum mechanical principles and methods in finding out interferometric behavior and allowed energy states of particles with arbitrary spins.

Unit – I: Electromagnetic Radiation and Interference: (4)

Expression for the electric field beyond Coulomb's law; The dipole radiator; Physics of interference – Two dipole radiator

Unit – II: Diffraction and Polarization: (4)

The resultant amplitude due to *n* equal oscillators; Diffraction Grating; The electric vector of light; Birefringence; Polarisers

Unit – III: Capacitance and Dielectrics: (4)

Electrostatic energy; Capacitance of a Parallel-Plate Capacitor; The dielectric constant; The polarization vector

Unit – IV: Special Relativity: (4)

The Lorentz transformation; Slowing of clocks; Contraction of length; Relativistic energy

Unit – V: Quantum Behaviour – I: Particles and Waves: (4)

Experiments with bullets, waves and electrons; The uncertainty principle

Unit – VI: Quantum Behaviour – I: The Magnetism of Matter: (4)

The Precession of atomic magnets; Angular momentum in Quantum Mechanics; The magnetic energy of atoms; Quantized magnetic states

Text Book:

R. P. Feynman, R. B. Leighton and M. Sands, **'The Feynman Lectures on Physics'**, *Pearson Education* (2006)

Reference Books:

1. J. Walker, D. Halliday, R, Resnick, **'Principles of Physics'**, Wiley *Student Edition* (10th Edition)

2. H. Young and Roger Freedman, **'University Physics'**, Pearson Addison Wesley (12th Edition)

BS1103 CHEMISTRY- I

Teaching Scheme: **Examination Scheme:**

Lectures: 2 Hrs/Week In-Semester: **50** Marks

Tutorial: 1 Hr/Week End-Semester: **50** Marks

Credits: 3

Course Objectives:

The Chemistry course is designed such that the learners develop a sound background of fundamental concepts and principles relevant in the engineering context. The course facilitates undergraduates to learn bonding theories, methods of analysis and evaluate role of chemical substances. They analyze chemical processes related to engineering applications. Also the course inculcates basic problem solving skills involving chemistry principles.

Course Outcomes:

1. State laws, formulae, definitions and properties.
2. Comprehend synthesis procedures and analytical methods in qualitative and quantitative estimation.
3. Apply principles of fundamental chemistry for solving problems.
4. Analyze chemical processes for engineering applications based on chemical reactions and evaluate the role of chemical substances.
5. Critique the effect of different parameters on the properties of chemical substance.

Unit – I: Chemical Bonding **(05)**

Types of bonds - primary & secondary types with examples, hybridization based on valence bond theory, VSEPR theory, molecular orbital theory with respect to bonding in homo and hetero nuclear diatomic molecules.

Unit – II: Water Analysis and purification **(06)**

Chemical Analysis of water hardness, alkalinity and effect of hard water in boilers, Internal

treatment of boiler feed water, water softening techniques (Permutit and Ion exchange method) and membrane based processes.

Unit – III: Electro chemistry **(06)**

(a) Fundamentals of an electrochemical cell, EMF of cell, reference and indicator electrodes, conductance in solution and conductometric titration.

(b) Battery Technology

Primary & secondary cell, battery characteristics, Ni-Cd cell, Lithium-ion battery, rechargeable batteries, Fuel cell technology.

Unit – IV: Instrumental methods of Analysis-I **(04)**

Basic principles,instrumentation and applications of pHmetry, Potentiometry, Chromatography

Unit – V: Coordination Chemistry **(05)**

Introduction,Classification of ligands, naming coordination compounds, Werner and Sidgwick theory, VBT, CFT for Td and Oh complexes. Applications and comparison of VBT & CFT.

Unit – VI: Photochemistry **(04)**

Photochemical reactions, Laws of Photochemistry and quantum yield, energy transfer in photochemical reaction, applications.

Text Books:

1. Arun Bahl & G.D. Tuli,**Essentials of Physical Chemistry,***S.Chand Publications* (2014)
2. S.S. Dara **'Engineering Chemistry'** *S. Chand Publications (2010)*
3. Puri, Sharma, Kalia **'Principles of Inorganic Chemistry'**: *Milestone Publications* (2009)
4. B.S. Chauhan **'Engineering Chemistry'** :*Univ Sc Press.*(third edition)2009
5. Shashi Chawla **'A Text Book Of Engineering Chemistry'**: *Dhanpat Rai & Co.*(2015)

6. Jain and Jain '**A Text Book Of Engineering Chemistry**' *Dhanpat Rai & Co.*
7. Gurdeep Chatwal **'Instrumental methods of Chemical Analysis'** *Himalaya publ.house*

Reference Books:

1. Steven S. Zumdahl, **‘Chemistry concepts and applications’,** *Cengage learning publication* (2009)
2. Ram D. Gupta,**'Hydrogen fuel'***C.R.C.Publications*(2009)
3. Puri, Sharma, Pathania **'Principles of Physical Chemistry'** *: Vishal Publ. Co.*(2015)
4. Robert D. Braun*' **Instrumental methods of analysis'** Pharmamed press* (2010)

ES1101 Basic Electrical and Electronics Engineering - I

Teaching Scheme: **Examination Scheme:**

Lectures: 3 Hrs/Week In-Semester: 50 Marks

Credits: 3 End-Semester:50Marks

Course Objectives:

1. To make students familiar with the fundamental concepts of electric and magnetic circuits.
2. To educate the students about the realization of basic theoretical concepts & laws in real physical world.
3. To educate the students about the construction and applications of diode
4. To educate the students about the construction and applications of BJT

Course Outcomes:

After completion of course, students will be able to

1) Determine energy consumption for electro-thermal and electro-mechanical systems as well as analyze the temperature effect on resistance
2) Analyze given magnetic circuit and find circuit parameters
3) Analyze given DC circuit and calculate its parameters
4) Calculate average value and RMS value of sinusoidal and non-sinusoidal AC waveforms.
5) Analyze I-V characteristics of semiconductor diodes and transistors and design simple analog circuits using these devices

Unit – I: Introduction to electrical systems (05)

Review of basic electrical terms, Effect of temperature on resistance, Resistance temperature coefficient, insulation resistance, Work, Power and energy calculations for thermal, mechanical and electrical systems.

Unit – II: DC Networks (07)

Kirchoff's laws, Mesh and Nodal Analysis, Thevenin , Norton and Superposition Theorems, maximum power transfer therom, Network Simplifications using star-delta / delta-star transformations.

Unit – III: Electromagnetism and Magnetic Circuits (06)

Magnetic field due to electric current, Force on a current carring conductor, Electromagnetic induction, direction and magnitude of induced EMF, magnetomotive force and magnetic field strength, relative and absolute permeability, reluctance, series and parallel magnetic circuits, magnetic materials and B-H curve, self and mutual inductance, coupling coefficient,energy stored in magnetic circuits.

Unit – IV: Electrostatics and AC fundamentals (06)

A. Electrostatic field, electric flux density, electric field strength, permittivity. Capacitor and capacitance, dielectric strength and breakdown voltage, capacitors in series and parallel,composite capacitors, energy stored in capacitors, charging and discharging of capacitors and time constant

B. Generation of alternating emf, waveform terms and definitions, average value and rms values for sinusoidal and non sinusoidal currents and voltages, phasor representation of an alternating quantity

Unit – V: Diodes and rectifiers (06)

Overview of Semiconductor physics and p-n junction theory, Junction diode, construction and characteristic of p-n junction diode, zener diode, LED, photodiode, Half wave, full wave and bridge rectifiers, need of capacitor filter, rectifier operation with capacitor filter, zener diode as a voltage regulator, block diagram of Regulated power supply

Unit – VI: Junction Transistor Amplifiers (06)

Bipolar junction transistor, Construction of BJT, Types of biasing:-fixed bias and self bias circuit, BJT characteristics for-CE,CB,CC configurations, relationship between α and β, load line for a transistor, application of transistor as a switch and amplifier.

Text Books:

1. Hughes, **'Electrical and Electronic Technology'**, *pearson education*, (9^{th} edition), (2009)

Reference Books:

1. D. P. Kothari and I.J. Nagrath, '**Basic Electrical Engineering**', *McGraw-Hill*, (3^{rd} edition), (2010)
2. A. E. Fitzgerald, A.Grabiel,'**Basic Electrical engineering**',*McGraw-Hill*, (5^{th} edition), (2009)
3. Floyd, '**Electronic Devices and Circuits**', *pearson education, (*7^{th} edition),(2008)

ES 1102 Fundamentals of Programming Languages - I

Teaching Scheme:
Lectures: 1 Hr/Week
Credits: 1

Examination Scheme:
In-Semester: 25 Marks

Course Objectives:

1. Learn the fundamentals of building blocks of computer.
2. Understand how to formulate the programming language statements from description of a problem in English.
3. Understanding of decision and iteration interpretation in a programming language.
4. Understand basic building blocks of simple website.

Course Outcomes:

Students will be able to

1. Write algorithm based on given problem statement
2. Draw flow chart for a given problem statement
3. Write the code for simple problem statement
4. Debug the code snippets manually

Unit – I: Introduction to Programming (02)

Introduction to computer, Anatomy of a computer: Hardware and software, Operating system, Types of programming languages: Machine language, Assembly language, High level languages, Selection of language, Algorithm: As a program, As a flow-chart, Pseudo code

Unit – II: Writing First C Program (02)

Structure of a C program, Writing C program, Introduction to library functions in C, Files generated in C program, Comments, Indentation

Unit – III: Variables and Operations (03)

C language variables: Numeric, Character, Declaring and Initializing variables, Constants: Integer, Floating point, Character, String Operators: Arithmetic, Relational, Equality, Logical, Unary, Conditional, Bitwise, Assignment, Comma, sizeof, Operator precedence variable scope: Local and Global scope, Type casting and conversion

Unit – IV: Control flow in C Language (03)

Conditional branching statements: if statements, if-else Statement, Switch case, Iterative statements: while loop, do-while loop, for loop, Nested loops, break and continue statements

Unit – V: Arrays (02) Introduction to Arrays, Accessing Array elements, Internal representation of Arrays in C, Working with one-dimensional array, Introduction to two-dimensional arrays

Unit – VI: Introduction to Website Development (02)

Introduction to blogging and WordPress : Creating a simple website, Content creation, Pages and Blogs, Page linking, Comments, Adding contents like Multimedia, Presentations, Themes

Text Books:

1. Reema Thareja, **'Introduction to C programming',** *Oxford University Press* (2nd edition), (2015)
2. Pradeep Day, '**Computer Fundamentals and programming in C'**, *Oxford University Press*, (2nd edition) (2013)

Reference Books:

1. B Kernighan, D Ritchie, '**C programming Language'**, *Prentice Hall Software Series*, (2nd edition) (1988)

ES1103 Engineering Graphics

Teaching Scheme: **Examination Scheme:**

Lectures: 2 Hrs/Week In-Semester: 25 Marks

Credits: 2

End-Semester: 25Marks

Course Objectives:

a) To apply theory of projections and standard conventions in engineering drawing.

b) To understand the methods to draw various engineering curves.

c) To develop the visualization and interpretation skills, for the physical objects.

d) To develop free hand sketching skills.

Course Outcomes:

After completing the course students will be able to draw

a) Orthographic projections of an object.

b) Engineering curves by applying the given method.

c) Isometric views and development of surfaces of the given object.

4. Free hand sketches of simple machine elements.

Unit – I: Introduction to Engineering Drawing (02)

Layout and sizes of drawing sheets, drawing instruments, types of lines used in drawing practice, dimensioning systems, representation of tolerances, standard codes by B.I.S (SP-46).

Unit – II: Curves in Engineering Practice (05)

Construction of ellipse, parabola, hyperbola, involute, cycloid, archimedean spiral, helix on cone and cylinder.

Unit – III: Orthographic Projections (08)

Theory of projections, methods of obtaining orthographic views,sectional orthographic projections.

Unit – IV: Isometric Projections (08)

Isometric axes, Isometric scale, isometric projections and views, construction of isometric view from given orthographic views.

Unit – V: Development of lateral surfaces of solids (05)

Parallel line development, radial line development, methods to transfer points for development of prisms, pyramids, cylinder and cone.

Unit – VI: Free hand sketching (02)

Free hand sketching of front view and/or top view of standard machine elements –thread forms, hexagonal headed bolt and nut, screws, shaft and keys, spring, welded and riveted joint.

Text Books:

1. N. D. Bhatt and V. M. Panchal,**'Engineering drawing, plane and solid geometry'**, *Charotor Publication House.*

a) R. K. Dhawan,**'A text book of Engineering Drawing'**, *Pearson Education Inc*.

b) P.S. Gill, **'Engineering Graphics'**, *Kataria and sons Publications.*

c) M.L.Dabhade, **'Engineering Graphics'**, *Vision Publications.*

Reference Books:

a) Warren J. Luzzader, **'Fundamentals of Engineering Drawing'**, *Prentice Hall of India, New Delhi.*

b) Fredderock E. Giesecke, Alva Mitchell, **'Principles of Engineering Graph-**

ics',*Maxwell McMillan Publishing.*

c) Dhananjay A. Jolhe, **'Engineering Drawing'**, *Tata McGrawHill Publishing Co. Ltd.*

ES 1104 Environmental Studies

Teaching Scheme:

Lectures: 2Hrs/Week

Tutorial: 1Hr/Week

Credits: 3

Examination Scheme:

In-Semester: 50 Marks

End-Semester: 50 Marks

Course Objectives:

1. It is an interdisciplinary approach to understand environment.
2. It enhances the ability to understand Environmental Problems.
3. Understand the relevance and importance of natural resources in the sustenance of life on earth and living standard.
4. To develop the ability and understand role of Individual in Environmental Protection

Course Outcomes:

A student should be able to obtain/develop:

1. Develop an understanding of environmental pollutions and hazards due to engineering/technological activities and general measures to control them.
2. Analyse the relationships between environmental laws across multiple sectors (local, state, national and international) Comprehend the importance of ecosystem and biodiversity.
3. Develop an understanding of different natural resources including renewable and non-renewable resources.

4. Identify suitable controlling measures for different types of solid wastes.

5. Improve fundamental knowledge of the inter-relationships between the built environment and natural environment.

6. Discuss an action plan for sustainable alternatives that integrate science, humanities and social perspective

Unit – I: Introduction (05)

Concept of environment and multidisciplinary nature of environmental studies:

a) Definition of Environment, multidisciplinary nature of Environmental Studies, scope, importance of Environment, Public awareness for Environment

b) Concept, Ecosystem characteristics:-Biotic abiotic, functional attributes

c) Energy flow in ecosystem: - Universal and single channel energy flow model, Nutrient Cycling:- Nitrogen cycle, carbon cycle, phosphorus cycle,

d) Concept of biodiversity

Unit – II: Integrated built environment (05)

d) Concept of integrated built environment – natural & man-made.

e) Eco-friendly materials in construction - Introduction, sources, Classification,properties and materials.

f) Principles of Building Planning: - Aspect, prospect, grouping, privacy, roominess, sanitation, orientation, circulation, elegance, economy.

g) Building bye laws (concept):- Building line, control line, set back distance, F.S.I., Built up area.

h) Concept of green building, advantages of green building, Introduction LEED rating system.

Unit – III: Renewable and Non- Renewable resources and it's Conservation (04)

f) Natural resources: Types of Renewable- Forest, water - causes of depletion, Conservation

g) Non-renewable resources, types, method of harnessing energy

Unit – IV: Environmental Pollution (05)

g) Introduction, Classification of pollution - Air and water - sources, causes, effects & remedial measures.

h) Solid waste generation, Collection of solid wastes, processing techniques, E- waste generation and methods of disposal.

i) Role of an individual in prevention of pollution.

Unit – V: Social Issues and Environment (05)

g) Unsustainable to sustainable development, urban problems related to energy, Climate change, global warming, acid rain, ozone layer depletion

h) Water conservation and Rain water harvesting

i) Introduction to Environmental Impact Assessment - Definition, introduction of methods with the help of a case study

j) Environment Protection Act, Forest Conservation Act, Public awareness.

Unit – VI: Smart City (03)

Concept and features of smart city, challenges of urbanization, selection process, strategy

Text books:

5. D.L. Manjunath, 'Environmental Studies', Pearson Education.

6. ErachBharucha, 'Text Book of Environmental Studies', UGC, Universities Press

Reference books:

5. D.K. asthana ,MeeraAsthana, ‘A Text Book Of Environmental Studies’,S.Chand.

6. Dr. J.P. Sharma, ‘Environmental Studies’,University Science Press.

7. Dr. Suresh K. Dhalmeja,‘Environmental Studies’,S.K.Kataria& Sons.

8. Anubha Kaushik, C.P.Kaushik, ‘Perspectives in Environmental Studies’,

New Age International Publishers.

9. Shah, Kale, Patki, ‘Building planning and Built environment’,

Tata McGraw Hill

10. Bukhootsow, ‘Energy policy and planning’,B- Prentice Hall of India New Delhi

BS1104 Physics and Chemistry Lab – I

Teaching Scheme	Examination Scheme
Practical: 2 Hrs/Week	In-Semester: 25
Credits: 1	

1: **Record** the observations as per the least counts of measuring instruments and carry out plotting and necessary calculations pertaining to the optical, electromagnetic and thermal systems.
2: Analyze the plotted data and experimental findings with the corresponding theoretical physical models pertaining to the optical, electromagnetic and thermal systems.
3: Analyze the sources of errors and arrive at conclusions pertaining to the behavior of optical, electromagnetic and thermal systems
4: Determine quality parameters of water such as hardness, alkalinity etc
5: Use of instrumental techniques in quantitative estimations like conductometry, pH metry, potentiometry.
6: Select appropriate quantitative analysis for estimation of different parameters of the substance.
7: Interpret the significance of a technique and specific role of reagent in qualitative and quantitative analysis.

List of Experiments:

Physics
1. Michelson Interferometer
2. Specific heat of substance
3. Hall Effect
4. Balmer Series and Emission Spectra
5. Zeeman Effect (Demo)

Chemistry
1. Qualitative & quantitative Analysis of alkali /alkaline earth metals using Flame Photometry.
2. Colorimetric verification of Beer-Lambert's law.
3. Determination of molecular weight of polymer using Ostwald Viscometer.
4. Proximate analysis of coal.

ES 1105 Basic Electrical and Electronics Engineering Lab-I

Teaching Scheme:	**Examination Scheme:**
Practical: 2 Hrs./Week	Practical Exam: 25 marks
Credits: 1	

Course Outcomes:

After completion of course, students will be able to

1. Perform basic domestic wiring
2. Apply circuit laws to find the parameters of given electrical network
3. Build a basic regulated DC power supply
4. Analyse the performance of Transistor in CE configuration
5. Write techinical report of conducted experiment

List of experiments:

1. Study of different electrical and electronics components and instruments.
2. To perform electrical wiring to control lamps using one way and two-way switches.
3. Determination of Temperature Rise of a Medium Resistance
4. Verification of kirchoff's laws & superposition theorems
5. Verification of Thevenin's theorem.
6. Performance analysis of half wave,full wave rectifier with center tap transformer and bridge rectifier with and without filter.
7. Performance analysis of three terminal IC voltage regulator
8. Determination of frequency responce of CE amplifier.

ES 1106 Fundamentals of Programming Languages Lab - I

Teaching Scheme:
Practical: 2 Hrs/Week
Credits: 1

Examination Scheme:
Practical: 25 Marks

Course Objectives:
Familiarize students with

1. Learn basics of C programming.
2. Learn to write C program for a given logical solution.
3. Learn to make validation checks at required places.
4. Learn to apply programming concepts to solve problems.

Course Outcomes:
Students will be able to

1) Write algorithm based on given problem statement
2) Apply appropriate programming constructs
3) Write program for simple problem statement
4) Test program for different inputs

Section 1 (any 08 assignments)

1. A) Write a C program to accept the length of three sides of a triangle and to test and print the type of triangle - equilateral, isosceles, right angled or none of these.

   B) Find out area, perimeter of a given trigonometric figure
2. Write a C Program to display the table of any given number
3. Write a C Program to reverse a given number
4. Write a C Program to find whether a given number is Armstrong number or not.
5. Write a C Program to calculate Simple Interest
6. Write a C Program to convert temperature from Celsius to Fahrenheit
7. Write a C program to display all the prime numbers between 1 to n
8. Write a C program to generate a series (like Fibonacci)
9. Write a C Program to display the numbers divisible by 7 in a given range(e.g. 11 to 90)
10. Write a C Program to accept a number and convert every digit into word and display it
11. Write a C Program for finding roots of Quadratic Equation
12. Write a C Program to find the greatest possible length which can be used to measure exactly the lengths 4m 95cm, 9m and 16m 65cm (Hint HCF)

Section 2 (any 02 assignments)

1. The traffic light at three different road crossings change after every 48, 72 and 108 sec, if they all change simultaneously at 8:20:00 hrs., then at what time will they again change simultaneously? (Hint : LCM)
2. The average of 25 results is 18. The average of first twelve of them is 14 and the average of last twelve of them is 17. Find the thirteenth result. (Hint Average).
3. The taxi fare is Rs. 14 for the first kilometer and Rs. 2 for each additional kilometer. What will be the

fare for 10 kilometers?(Hint: Arithmetic Progression)

4. Roma's mathematics test had 75 problems, i.e. 10 arithmetic, 30 algebra and 35 geometry problems. Although she answered 70% of the arithmetic, 40 %of algebra and 60% of geometry problems correctly she did not pass because she got less than 60% of the questions right. How many more questions she would have needed to solve to earn 60% of passing grade?(Hint Percentage.)
5. A radio is purchased for Rs. 490/- and sold for Rs.465.50. Find the loss percentage(Hint: Profit and Loss)
6. In how many ways can a cricket 11 be chosen out of a batch of 15 players?(Hint Permutation and Combination)
7. Write a C Program to accept a number and convert every digit into word and display it

Section 3 (study assignment)

Design and develop a small application using Wordpress

Text Books:

1. Reema Thareja, **'Introduction to C programming',** *Oxford University Press* (2nd edition), (2015)
2. Pradeep Day, '**Computer Fundamentals and programming in C'**, *Oxford University Press*, (2nd edition) (2013)

Reference Books:

1. B Kernighan, D Ritchie, '**C programming Language'**, *Prentice Hall Software Series*, (2nd edition) (1988)

ES1107 Engineering Graphics Lab

Teaching Scheme:

Practical: 2 Hrs/Week

Credit: 1

Examination Scheme:

Practical: 25 Marks

Course Objectives:

Students will be able to

1. Apply theory of projections and standard conventions in engineering drawing.
2. Understand the methods to draw various engineering curves.
3. Develop the visualization and interpretation skills for the physical objects.
4. Develop free hand sketching skills.

Course Outcomes:

After completing the course students will be able to

Identify applications of engineering curves and draw the curves.

Understand and draw orthographic projections and isometric views of an object.

Draw the development of lateral surfaces of solids.

Create free hand sketches of the machine elements.

I: Introduction to Engineering Drawing (01)

Drawing sheet layouts, drawing instruments, standard codes by B.I.S (SP-46)

II: Assignments and Drawing Sheets (12)

- Engineering Curves.
- Orthographic Projections
- Isometric Projections

- Development of surfaces of solids.
- Free hand sketching.

III: Introduction to computer aided drafting package (02)

Features and applications of computer aided drafting packages, basic operations, and various commands for drawing, dimensioning, editing, saving and plotting the drawings.

NC 1201 Value Education

Teaching Scheme: **Examination Scheme:**

Lectures: 1 Hr /Week In-Semester: Nil

Tutorial: Nil End-Semester: Nil

Credits: Nil

Course Objectives:

1. To make understand importance of values in human behavior.
2. To understand adjustments required in one self and others to uphold values in society.
3. To understand importance of values in Family Life.
4. To understand ethics required by professionals in work place.

Course Outcomes:

1. Students will appreciate importance of values in all walks of life.
2. To develop women professional with strong ethics and above all be a good human being.
3. To help students to develop their own value system and action plan based on it.
4. To understand the impact of the Moral role of students in nation building and being a responsible citizen.
5. Understand effects of Global issue like Terrorism, Environment, different cultures etc.

Unit – I: Values and Self Development (03)

Value Education – Definition - relevance to present day - Concept of Human Values - self introspection - Self esteem.

Unit – II: Family values (03)

Components, structure and responsibilities of family - Neutralization of anger - Adjustability- Threats of family life - Status of women in family and society - Caring for needy and el-

derly -

Time allotment for sharing ideas and concerns.

Unit – III: Ethical values (03)

Professional ethics - Mass media ethics- Advertising ethics -Influence of ethics on family life -
psychology of children and youth – Leadership qualities - Personality development.

Unit – IV: Social values (03)

Faith, service and secularism - Social sense and commitment -Students and Politics -Social awareness, Consumer awareness, Consumer rights and responsibilities - Redressal mechanisms

Unit – V: Effect of international affairs on values of life/ Issue of Globalization (03)

Modern warfare -Terrorism. Environmental issues - mutual respect of different cultures, religions
and their beliefs.

Text Books:

1. Chakraborty, S.K., **'Values and Ethics for Organizations Theory and Practice'**, *Oxford
University Press, New Delhi*, (2001)

Reference Books:

1. T. Anchukandam and J. Kuttainimathathil (Ed) 'Grow Free Live Free', ***Krisitu Jyoti***

F. Y. B. Tech. Semester –II									
Course Code	Course Title	Teaching Scheme Hours / Week			Examination Scheme			Marks	Credit
		Lecture	Tutorial	Practical	In Semester	End Semester	Practical / Oral		
BS 1201	Engg. Mathematics - II	3	1	0	50	50	0	100	4
BS 1202	Physics - II	2	1	0	50	50	0	100	3
BS 1203	Chemistry - II	2	1	0	50	50	0	100	3
ES 1201	Basic Electrical and Electronics Engg. - II	3	0	0	50	50	0	100	3
ES 1202	Fundamentals of Programming Language - II	1	0	0	25	0	0	25	1
ES 1203	Basic Mechanical Engg.	3	0	0	50	50	0	100	3
ES 1204	Engg. Mechanics	2	1	0	50	50	0	100	3
BS 1204	Physics and Chemistry Lab - II	0	0	2	25	0	0	25	1
ES 1205	Basic Electrical and Electronics Engg. Lab - II	0	0	2	0	0	25	25	1
ES 1206	Fundamentals of Programming Language - II	0	0	2	0	0	25	25	1
ES 1207	Engg. Mechanics Lab	0	0	2	0	0	25	25	1
ES 1208	Workshop Practice – I	0	0	2	0	0	25	25	1
	Total	**16**	**4**	**10**	**350**	**300**	**100**	**750**	**25**
	Grand Total	**30**			**750**			**750**	**25**

MKSSS's Cummins College of Engineering for Women Karvenagar, Pune-411052

DEAN ACADEMICS
MKSSS's Cummins College of Engineering for Women Karvenagar, Pune-411052

Principal
MKSSS's Cummins College of Engg. For Women, Karvenagar, Pune-52.

APPROVED BY
Governing Body Members
MKSSS's Cummins College of Engineering for Women Karvenagar, Pune-411052

BS1201 Engineering Mathematics-II

Teaching Scheme: **Examination Scheme:**

Lectures: 3 Hrs/Week In-Semester: 50 Marks

Tutorial: 1 Hr /Week End-Semester: 50 Marks

Credits: 4

Course Objectives:
Mathematics is a necessary path to scientific knowledge which opens new perspective of mental activity. Our aim is to provide sound knowledge of engineering mathematics to make the students think mathematically and strengthen their thinking power to analyse and solve engineering problems in their respective areas.

Course Outcomes: Students will be able to
1. Solve first order first degree DE, apply it to model and solve simple engineering problems like R-C circuit, conduction of heat etc.
2. Apply Beta, Gamma, Error function and Leibnitz's rule of DUIS to solve integration of univariate function
3. Identify the characteristics of the given function and trace the curve.
4. Integrate multivariate functions over the given region and apply the knowledge to find area, volume, mass, density etc.
5. Obtain Fourier series of given periodic function; Find nth harmonics for given data.

Course Contents:

Unit – I: First order first degree Differential Equation (07)

Definition, Order and degree of Differential Equation, Formation of differential equation, solutions of differential equation, Exact differential equation, Linear differential equation and equations reducible to these types.

Unit – II: Applications of Differential Equations (05)

Applications of differential equations to engineering problems: simple electrical circuits, applications of chemical engineering, applications of mechanical engineering and applications of physics.

Unit – III: Integral Calculus (07)

Special Functions:-Gamma Function, Beta Function, Error function. Differentiation Under integral sign (Leibnitz's rule). Curve tracing of Cartesian form, polar form.

Unit – IV: Multiple Integrals (08)
Transformation of Co-Ordinate systems Spherical, Polar and Cylindrical, Double and Triple integrals with limits, Double and Triple integrals without limits. Dirichlet's theorem.

Unit – V: Applications of Multiple Integrals (06)

Area of cartesian curves, Area of polar curves, Volume of solid, Mass of plane lamina, Mass of solid.

Unit – VI: Fourier Series and Harmonic Analysis (09)

Definition of Fourier series, Dirichlet's conditions, full range Fourier series, half range Fourier Sine series, half range Fourier Cosine Series, Practical Harmonic analysis and applications to problems in Engineering.

Text Books:

1. B.S. Grewal, 'Higher engineering Mathematics', Khanna publishers, Delhi (40th edition), 2008 .

2. B. V. Ramana, 'Higher Engineering Mathematics ',Tata McGraw Hill Publications, (2007)

Reference Books:

1. C.R.Wylie, L.C. Barrette, 'Advanced Engineering Mathematics', McGraw Hill Publications, New Delhi.(6th edition),(2003)

2. Peter V. O'neil, 'Advanced Engineering Mathematics', Thomson Brooks / Cole, Singapore (5th edition), (2007).

3. Erwin Kreyszig, 'Advanced Engineering Mathematics' ,Wiley Eastern Ltd.(8th Student Edition), (2004).

BS 1202 PHYSICS– II

Teaching Scheme:

Lectures: 2Hrs/Week

Tutorial: 1Hr/Week

Credits: 3

Examination Scheme:

In-Semester: 50 Marks

End-Semester: 50 Marks

Course Objective:

The objective of this course is to provide an 'algorithmic' introduction of the basic principles of Quantum Physics to the first year students of engineering. Throughout the course, the applications of Quantum Physics will be discussed by emphasizing the laws of combining 'probability amplitudes'. This will be done through several case studies and experimental situations.

Course Outcomes:

By taking this course, the learner will be able to –

1: **Apply** the laws of combining probability amplitudes for obtaining intensity distributions of ensembles of identical microscopic systems.
2: Differentiate between domain – specific nature of probability amplitudes in elementary quantum mechanical situations.
3: Justify the use of the laws of combining probability amplitudes in situations involving photons and two – state and multi – state quantum systems.

Unit – I: Probability Amplitudes: (4)

The laws for combining amplitudes; The two-slit interference patter; Scattering from a crystal

Unit – II: Identical Particles: (4)

Bose particles and Fermi particles; Case studies involving use of the exclusion principle

Unit – III: The Dependence of Amplitudes on Time: (4)

Stationary states; Potential energy and energy conservation; The precession of a spin-half particle

Unit – IV: The Hamiltonian Matrix: (4)

Resolving state vectors; How state changes with time; Hamiltonian Matrix

Unit – V: Two-state Systems and Single Qubit Logic Gates: (4)

Experiments with bullets, waves and electrons; The uncertainty principle

Unit – VI: Band Theory of Solids and Semiconductor Physics: (4)

States for an electron in a lattice; Electrons and holes in semiconductors; The Hall effect; Rectification at a semiconductor junction; The transistor

Text Book:

R. P. Feynman, R. B. Leighton and M. Sands, **'The Feynman Lectures on Physics'**, *Pearson Education* (2006)

Reference Books:

1. J. Walker, D. Halliday, R, Resnick, **'Principles of Physics'**, Wiley *Student Edition* (10^{th} Edition)
2. H. Young and Roger Freedman, **'University Physics'**, Pearson Addison Wesley (12^{th} Edition)

BS-1203 Chemistry II

Teaching Scheme:

Lectures: 2 Hrs/Week

Tutorial: 1 Hr/Week

Examination Scheme:

In-Semester: 50 Marks

End-Semester: 50 Marks

Credits: 3

Course Objectives:

The Chemistry course is designed for the learners to develop a sound background of fundamental concepts and principles relevant in the engineering context. The course facilitates undergraduates to evaluate the role of chemical substances in different methods of preparation and analysis.They analyze chemical processes related to engineering applications. Also the course inculcates basic problem solving skills involving chemistry principles.

Course Outcomes:

By taking this course, the students will be able to

CO1: Apply spectral and analytical techniques for chemical analysis.

CO2: State laws, definitions and identify physical parameters affecting composition of systems.

CO3:Elucidate on structure and synthesis of materials.

CO4: Evaluate types, factors, mechanisms related to corrosion and its preventive methods.

CO5: Analyze materials for their properties and applications such as fuel or speciality materials.

Unit – I: Instrumental methods of Analysis II (04)

Basic principles, theory, instrumentation and applications of Uv-Vis Spectrophotometry; Flamephotometry.

Unit – II: Polymer Chemistry (07)

Basic terms, molecular weight determination, types of polymerization and its mechanism (free radical and ionic), compounding of plastics, Speciality polymers, Recycling of polymers

Unit – III: Chemistry of fuels (09)

Calorific value, Bomb & Boys' calorimeter, Proximate and Ultimate analysis of coal, Crude

oil: refining, knocking, alternate fuels, rocket propellants, Combustion: calculation of air required for combustion.

Unit – IV: Corrosion **(04)**

Dry and wet corrosion mechanism, types, factors affecting corrosion, Protection against corrosion: Cathodic and anodic protection, powder coating and metallic coatings.

Unit – V: Phase Rule **(03)**

Gibbs Phase Rule, one Component system- Water system, Sulphur system, Two component system- (Pb-silver alloy). Applications and limitations of phase rule.

Unit – VI: Nanomaterials **(03)**

Introduction to nanomaterials, synthesis by top down and bottom up methods, properties and typical applications of nanomaterials.

Text Books:

1. Arun Bahl and G.D. Tuli, **'Essentials of Physical Chemistry'**, (2014/2016)
2. S.S. Dara **'Engineering Chemistry'** *S.Chand Publications* (2010)
3. Puri, Sharma, Kalia **'Principles of Physical Chemistry'** *Milestone Publication* (2009)
4. B.S. Chauhan **'Engineering Chemistry'** *Univ Sc Press.*(2015)
5. Shashi Chawla **'A Text Book Of Engineering Chemistry'** *Dhanpat Rai & Co.* (2015)
6. S.K. Kulkarni **'Nanotechnology: principles and practices'** (2014)
7. Gurdeep Chatwal **'Instrumental methods of Chemical Analysis'** *Himalaya publishing house* (1996)

Reference Books:

1. Ram D. Gupta,**'Hydrogen as a fuel'** *C.R.C.Publication* (2009)
2. Puri,Sharma,Pathania **'Principles of Physical Chemistry'** *Vishal Publishing Co.* (2015-16)
3. **Robert D. Braun 'Instrumental methods of analysis'** ***Pharmamed press*** **(2010)**

ES 1201 Basic Electrical and Electronics Engineering – II

Teaching Scheme:
Lectures: 3 Hrs/Week
Credits: 3

Examination Scheme:
In-Semester: 50 Marks
End-Semester: 50 Marks

Pre-requisite : Semiconductor physics

Course Objectives:
1. To make students familiar with the fundamental concepts of AC circuits
2. To familiarize the students with three phase supply
3. To develop a clear understanding of operation and application of transformer
4. To make students familiar with Digital Circuits
5. To introduce Basics operational amplifier (IC 741) and its applications

Course Outcome:
Having successfully completed this course, the student will be able to:
1. Analyze and determine parameters of single phase AC circuit.
2. Quantify parameters of single phase transformer related to its operation and use .
3. Develop applications of logic gates for building combinational and sequential circuits.
4. Build simple linear and non-linear circuits using operational amplifier.
5. Analyze characteristics of different power devices and transducers.

Unit I: **AC Circuits** **(08)**
Behavior of pure R,L,C in ac circuits,Series and parallel RL, RC and RLC circuits, concept of Impedance and admittance, power triangle and power factor.Resonance in series and parallel RLC circuit, Three phase voltage generationand waveform, star and delta balanced systems. Relationship between phase and line quantities, phaser diagram, power in a three phase circuit.

Unit II : Single phase Transformers **(07)**
1 Φ transformer: concept, types, working, ideal transformer, practical transformer, equivalent circuit, phasor diagram, efficiency and regulation calculations. Introduction to three phase transformer.

Unit III: Digital Electronics **(07)**
Binary number systems and binary arithmetic, basic gates, implementation of basic gates using universal gates, Boolean algebra, standard representation of logic functions (SOP and POS forms), Introduction of Combinational logic circuits like multiplexer ,demultiplexer ,half adder and full adder, Introduction of Sequential logic circuits like flip- flops (SR, D), counters and shift registers.

Unit IV: OPAMP **(07)**
Introduction to operational amplifiers, opamp configurations, modes and parameters, Negative feedback concept and applications like comparators, summing amplifiers, integrators and differentiators.

Unit V: POWER DEVICES **(07)**

Construction, characteristics and turn on mechanism of SCR, two transistor analogy of SCR, concept of line and forced commutation. Introduction to phase control concept. Construction, characteristics of IGBT and MOSFET.

Unit VI: Transducers **(06)**

Introduction to Transducers, selection of transducers, classification of transducers. Types of transducers such as LVDT, RTD, Thermistor and strain gauge.

Text Books:-

Hughes,"Electrical & Electronic Technology", *Pearson Education, 9th Edition*

Reference Books:-

1. AP Malvino & Donald Leach,"Digital Principles and Applications", *McGraw Hill Education, 4th edition*

2. Floyd ,"Electronic Devices and Circuits", *Pearson Education India, 8th edition*

3. H.S. Kalsi "Electronic Instrumentation",*TMH publication,2nd edition*

4. Jacob Millman & C C Halkais, Chetan parikh,"Integrated Electronics", *TMH, 2nd edition*

5. D.P. Kothari and I.J. Nagrath, "Basic Electrical Engineering", *Tata McGraw- Hill, 3rd Edition.*

ES 1202 Fundamentals of Programming Languages - II

Teaching Scheme:
Lectures: 1 Hr/Week
Credits: 1

Examination Scheme:
In-Semester: 25 Marks

Course Objectives:
Familiarize students with

1. Understand role of functions and it's utility in programming.
2. Understand the use of pointers in memory management.
3. Understand the utility of need and utility of user defined data types.
4. Learn and explore mobile application development environment.

Course Outcomes:
Students will be able to

1. Write program using functions
2. Write code for effective memory management
3. Write code using appropriate user defined data types for various applications
4. Write code with user defined functions similar to inbuilt functions

Unit – I: Functions in C (03)
Concept of Function, Function declaration, Function definition, Function Call, Return statement, Passing parameters: Call by value, Recursion

Unit – II: Strings (02)
Introduction, Reading Strings, Writing Strings, Strings Operations: Counting characters in String, Converting into upper case and lower case, Concatenation, Appending, Comparing, Reverse

Unit – III: Introduction to Pointers in C (02)
Understanding Computer memory, Introduction to Pointers, Declaring pointer variable, Function Call by reference, Pointer and Arrays, Role of Pointers in Passing an Array to a Function, Pointers and Strings

Unit – IV: Structures (02)
Introduction to Structures: Declaring Structure and Structure Variables, Initializing Structure, Accessing members of Structure

Unit – V: Unions, Enumeration Data types (02)
Declaring Union and its members, Accessing members of Union, Enumeration Types

Unit – VI: Mobile application Development (02)
Introduction, Web apps vs. Native apps, Introduction to mobile operating System like Android / IOS / Windows, Features and architecture of Mobile Operating System, Generating GUI and views, Layouts and Application Components, Creating simple mobile application.

Text Books:

1. Reema Thareja, **'Introduction to C programming',** *Oxford University Press* (2nd edition), (2015)
2. Pradeep Day, '**Computer Fundamentals and programming in C'**, *Oxford University Press*, (2nd edition) (2013)

Reference Books:

1. B Kernighan, D Ritchie, '**C programming Language'**, *Prentice Hall Software Series*, (2nd edition) (1988)

ES1203 Basic Mechanical Engineering

Teaching Scheme:

Lectures: 3Hrs/Week

Credits: 3

ExaminatioScheme:

In-Semester: 50 Marks

End-Semester: 50 Marks

Course Objectives:

a) To provide an overview of mechanical engineering systems (Power plant, Manufacturing plant, Maintenance systems, transmission systems).

b) To enable students to understand terminology used in Mechanical engineering with its significance.

c) To make student understand concept of Mechatronics System.

Course Outcomes:

a) The student will be able to differentiate between major areas like Design, Manufacturing and Thermal in mechanical industries while addressing a problem.

b) The student will be able to select an appropriate sector while finding solution to a problem.

c) The student will be aware of avenues available while choosing career opportunities in mechanical engineering Industry.

d) Understand the underlying principle of energy conversion systems and power plants, power producing and Power absorbing devices.

e) Students will be able to identify Mechatronics System and its components.

Unit – I: Introduction to basic mechanical engineering (06)

Industry overview-Comparison between process, product and service industry. Work environment for Mechanical industries, role of a mechanical engineer, ethics, professional hazards and safety concerns in mechanical industry. Typical manufacturing method of a product.

Unit – II: Introduction to thermal engineering (08)

Thermodynamic system, properties, states, process, cycle, first law of thermodynamics, application of first law to open and closed systems,second law of thermodynamics, conceptual difference between heat engine, heat pump and refrigerator, significance of efficiency and co-efficient of performance. Numerical on appropriate topics.

Unit – III: Power producing devices and power absorbing devices (08)

Power producing devices-Internal combustion engines and turbines, power plants.

Power absorbing devices-Centrifugal pumps, reciprocating units, vapour compression refrigeration, air conditioning systems.

Energy management system-fluctuations in demand-supply of energy,need of power grid, concept of energy audit.

Unit – IV: Introduction to design engineering

Introduction to engineering materials, elements and principles of engineering design, basic procedure, Basic requirement, standards in design, aesthetic and ergonomic considerations in design.

Basic machine elements, shaft, key, coupling, bearing, clutch and brake.

Mechanical drives, belt, chain and gear.

Unit – V: Introduction to manufacturing (08)

Operation on different machine tools, lathe, Milling, Drilling.

Joining of metals, welding-gas and arc, TIG, MIG, Soldering, brazing.

Hot and cold working-Forging, rolling, extrusion.

Unit – VI: Introduction to Mechatronics (06)

Definition(S) of Mechatronics, Mechatronics system Components, Levels of Mechatronics system, Examples of Mechatronics (products and systems in manufacturing), Advantages of Mechatronics with Traditional Systems.

Text Books:

a) C.P. Aurora, '**Thermodynamics'**, *Tata McGraw Hill education*, (2001).

b) BasantAgarwal, C.M Agarwal, '**Basic Mechanical Engineering'**, *Wiley Ind. Pvt. Ltd*.

c) V B Bhandari, **'Design of Machine Elements'**, *Tata McGraw Hill*, (2nd edition), (2007).

d) S. K.HajraChoudhury, S.K.Bose, A.K.HajraChoudhury, **'Elements of workshop technology, volume I and II',** *Media promoters and publishers pvt. Ltd*(7th edition).

e) W.Bolton, '**Mechatronic-a multidisciplinary approach',** *Prentice Hall*, (4th edition), (2009).

f) Class room notes.

Reference Books:

a) Moran, Shapiro, Boettner, Bailey, '**Principles of engineering thermodynamics'**, *Wiley*, (7^{th} edition).

b) Rayner Joel, '**Basic engineering thermodynamics'**, *Addison-Wesley*, (5^{th} edition).

c) Y. A. Cengel and M. A. Boles, '**Thermodynamics, an Engineering Approach**', (4th edition).

d) S.S. Rattan, '**Theory of Machine'**, *McGraw Hill*, (4^{th} edition).

e) B.S. Raghuwanshi, '**A course in workshop technology'**, *DhanpatRai&co*.

f) Kalpakjian, Schmid, **'Manufacturing engineering and technology**', *Pearson*, (4^{th} edition).

7. Nptel course112105127/1, 112105127/2

ES 1204 Engineering Mechanics

Teaching Scheme:

Lectures: 2Hrs/Week

Tutorial: 1Hr/Week

Credits: 3

Examination Scheme:

In-Semester: **50** Marks

End-Semester: **50** Marks

Course Objectives:

1. To develop the ability of students to analyze any problem in a simple and logical manner.
2. To make the students understand the fundamental principles of mechanics which are the foundation of much of today's engineering.
3. To develop logical thinking of the students for application in engineering.
4. To provide an introduction to the basic quantities of mechanics.

Course Outcomes:

A student should be able to obtain/develop:

1. An ability to apply knowledge of mathematics, science and engineering

2. A recognition of the need for, and an ability to engage in, life-long learning.

3. Application of Newton's laws of motion

4. Knowledge of kinematic & kinetic analysis.

Unit – I: Introduction to Statics (06)

1. Fundamental concepts and principle (The parallelogram law of addition of forces, the principle of transmissibility, Newton's laws of motion**,** Newton's law of gravitation). Introduction to a force in a plane, Types of force system, resolution & composition of forces, Methods of composition to find resultant, moment of force, Varignon's theorem, couple, equivalent force couple system.
2. Introduction to force in a space, problems on resultant of concurrent force system
3. Equilibrium- Introduction to concept of equilibrium, Conditions of equilibrium, Free body diagram, equilibrium under different forces, equilibrium of concurrent parallel & general forces in a plane.

Unit – II: Introduction to type of Supports and Beam (05)

1. Types of supports (Fixed, roller, hinged support)

   Types of loads on a beam (point load, uniformly distributed load, uniformly varying load)
   Types of beams (simple beam, cantilever beam, compound beam)

2. Problems on Reactions & analysis of beams
3. Centroid- Definitions (Center of gravity of two dimensional body, center of mass, centroid), procedure to find centroid of regular plane lamina.

Unit – III: Introduction to Friction (03)

Definition and classification of friction, coefficient of static and kinetic friction ,angle of friction, angle of repose, problems on block friction and ladder friction

Unit – IV: Rectilinear Motion (05)

2) Variables in Rectilinear motion- Time, Position, Displacement, Distance travelled, Velocity, Acceleration

   Equations of motion for constant acceleration & motion under gravity, variable acceleration, relative motion based on kinematic equations.

3) Application of Newton's second law of motion for rectangular co-ordinate system (D' Alembert's principle)

Unit – V: Curvilinear Motion (05)

1) Equation of motion in rectangular components, Normal & Tangential components, Radial & Transverse components.

2) Projectile motion- Definition and derivation (time of flight, horizontal range, angle of projection, maximum height, trajectory),Projectile on horizontal plane only

Unit – VI: Work Energy Principle (04)

1. Introduction and definition of Work, power, energy, conservative & non- conservative forces, Conservation of energy, work-energy principle.

2. Problems on Work done by different forces (External force, Frictional force, Gravitational force, Spring force).

Text books:

1. A Nelson, '**Engineering Mechanics Statics and Dynamics**', *Mc Graw Hill Education.*

2. R.S. Khurmi, '**A Textbook of Engineering Mechanics**', *S. Chand & Company Ltd.*

Reference books:

- Beer & Johnson, '**Vector mechanics for engineers**', *Mc Graw hill publication.*
- I. H. Shames & G.K.M. Rao, '**Engg. Mechanics'**, *Pearson.*
- R. C. Hibbler, '**Engg. Mechanics statics & dynamics**', *Pearson publication*
- **S. Timosenko, DPT.young & J.V.Rao, 'Engineering mechanics',** ***Tata Mc Graw hill education pvt. Ltd. New delhi.***

BS 1204 Physics Chemistry Lab – II

Teaching Scheme:

Lectures: 2Hrs/Week

Tutorial: 1Hr/Week

Practical: 2 Hrs/Week

Credits: 1

Examination Scheme:

In-Semester: **25** Marks

1: **Record** the observations as per the least counts of measuring instruments and carry out plotting and necessary calculations pertaining to solid state physics, atomic and molecular system.
2: Analyze the plotted data and experimental findings with the corresponding theoretical physical models pertaining to solid state physics, atomic and molecular system.
3: Analyze the sources of error and arrive at conclusions pertaining to the behavior of solid state physics, atomic and molecular system.
4: Determine the molecular weight of a given polymer by viscometry.
5: Evaluate a solid fuel sample for its quality by proximate analysis.
6: Implement spectral analysis for a given chemical compound.

List of Experiments:

Physics

1. Michelson Interferometer
2. Specific heat of substance
3. Hall Effect
4. Balmer Series and Emission Spectra
5. Zeeman Effect (Demo)

Chemistry

1. Qualitative & quantitative Analysis of alkali /alkaline earth metals using Flame Photometry.
2. Colorimetric verification of Beer-Lambert's law.
3. Determination of molecular weight of polymer using Ostwald Viscometer.
4. Proximate analysis of coal.

ES 1205 Basic Electronics and Electrical Engineering Lab- II

Teaching Scheme: **Examination Scheme:**

Laboratory: 2 Hrs/Week End-Semester:25 Marks

Credits: 1

Pre-requisite : Instruments ,Electronics and electrical components,semiconductor physics.

Course Objectives:

3. To make students familiar with the fundamental concepts of single phase AC circuits
4. To make students familiar with three phase supply
5. To demonstrate working of single phase transformer
6. To explain combinational logic circuits
7. To introduce Basics operational amplifier (IC 741) and its applications

Course Outcome:

Having successfully completed this course, the student will be able to:

3. Apply fundamental concepts of single phase and three phase AC circuits.
4. Test performance parameters of single phase transformers.
5. Implement basic analog and digital circuits.
6. Verify characteristics of SCR and transducer.

List of Practicals:-

1. Performance analysis of L-C-R series circuit .
2. Load test on single phase transformer for determination of voltage regulation.
3. Performance analysis of 3 phase AC circuit.
4. Analysis of summing amplifier and difference amplifier using OPAMP.
5. Design and implementation of half adder and full adder circuits.
6. Illustrate effect of variation of displacement on output voltage of LVDT.
7. Verification of static characteristics of SCR.
8. Soldering Techniques (any small circuit like clippers, clamper, circuits using basic gates).

ES 1206 Fundamentals of Programming Languages Lab - II

Teaching Scheme:
Practical: 2 Hrs/Week
Credits: 1

Examination Scheme:
Practical: 25 Marks

Course Objectives:
Familiarize students with

1. Learn and acquire art of computer programming.
2. Learn advanced C programming features.
3. Learn to write C program for a given logical solution.
4. Learn to apply programming concepts to solve simple problems using arrays, functions and structures.

Course Outcomes:
Students will be able to

1. Write program using functions for given problem statement.
2. Write code using sequential memory management
3. Apply appropriate user defined data types for given statement.
4. Write program with user defined functions similar to library functions.

Section 1 (any 07 assignments)

1. Write a C program to swap 2 integers using user defined functions (call by value, call by reference).
2. Write a program in C to compute the factorial of the given positive integer using recursive function.
3. Write functions to convert feet to inches, convert inches to centimeters, and convert centimeters to meters. Write a program that prompts a user for a measurement in feet and converts and outputs this value in meters. Facts to use: 1 ft = 12 inches, 1 inch = 2.54 cm, 100 cm = 1 meter.
4. Write a menu driven program to perform following operations using Array of integers like (accept, display, print alternate number, sum of all numbers, search a number).
5. Write a program in C to sort n integers using bubble sort.
6. Write a menu driven program to perform string operations using library functions.
7. Write a menu driven program to perform string operations using user defined functions.
8. Define an integer pointer array of 10 integers. Initialize them to any integer values from the keyboard. Find the sum, average, minimum, and maximum of these 10 integers. Sort the 10 integers in descending order.
9. Write a program in C to compute addition / subtraction / multiplication of two matrices. Use functions to read, display and add / subtract / multiply the matrices.
10. For a class an examination is conducted and the results for the students of all the 5 subjects are recorded. Write C program to display the record of students. On the basis of the record compute

11. Write a menu-based program in C that uses a set of functions to perform the following operations:

    i. reading a complex number
    ii. writing a complex number
    iii. addition of two complex numbers
    iv. Iv.subtraction of two complex numbers
    v. multiplication of two complex numbers
    vi. Represent the complex number using a structure.

12. Write a C program to create an employee database using structure and perform operations such as accept, display, search by name, search by number, update a record.

Section 2 (any 02 assignments)

1. A string is provided from the user. Calculate the total number of characters in the string and the total number of vowels in the string with the number of occurrence in the string

2. College library has n books. Write C program to store the cost of books in array in ascending order.
   Books are to be arranged in descending order of their cost

3. Write a recursive function to obtain the first 25 numbers of a Fibonacci sequence. In a Fibonacci sequence the sum of two successive terms gives the third term. Following are the first few terms of the Fibonacci sequence:1 1 2 3 5 8 13 21 34 55 89

4. A factory has 3 division and stocks 4 categories of products. An inventory table is updated for each division and for each product as they are received. There are three independent suppliers of products to the factory:

   (a) Design a data format to represent each transaction

   (b) Write a program to take a transaction and update the inventory

   (c) If the cost per item is also given write a program to calculate the total inventory values.

5. Write a program that compares two given dates. To store date use structure say date that contains three members namely date, month and year. If the dates are equal then display message as "Equal" otherwise "Unequal".

6. Create a structure to specify data of customers in a bank. The data to be stored is: Account number, Name, Balance in account. Assume maximum of 200 customers in the bank.

   (a) Write a function to print the Account number and name of each customer with balance below Rs. 100.

   (b) If a customer request for withdrawal or deposit, it is given in the form: Acct. no, amount, code (1 for deposit, 0 for withdrawal) Write a program to give a message, “The balance is insufficient for the specified withdrawal”

7. An automobile company has serial number for engine parts starting from AA0 to FF9. The other characteristics of parts to be specified in a structure are: Year of manufacture, material and quantity manufactured.

Section 3 (study assignment)

Students should design and develop a small Android application for mobile.

Text Books:

1. Reema Thareja, **'Introduction to C programming',** *Oxford University Press* (2nd edition), (2015)
2. Pradeep Day, '**Computer Fundamentals and programming in C'**, *Oxford University Press*, (2nd edition) (2013)

Reference Books:

1. B Kernighan, D Ritchie, '**C programming Language'**, *Prentice Hall Software Series*, (2nd edition) (1988)

ES1207 Engineering Mechanics Lab

Teaching Scheme:

Lectures: 2 Hrs/Week

Tutorial: 1 Hr/Week

Credits: 1

Examination Scheme:

In-Semester: 25 Marks

No. of Experiments:

Part A-Experiments (any 7 experiments)

1. Verification of law of polygon of forces.
2. Verification of Varignon's theorem.
3. Verification of Lami's theorem.
4. Support reactions of simple beam.
5. To determine forces in space force system.
6. Study of Curvilinear motion.
7. Determination of coefficient of restitution.
8. To compare coefficient of friction of various pair of surfaces in contact.

Part B- Graphical analysis -(Any one)

1. To find resultant of force system.
2. To find support reactions of simple beam.

ES 1208 Workshop Practice I

Teaching Scheme:

Practical: 2 Hr/Week

Credit: 1

Examination Scheme:

Practical/Oral Examination: 25 marks

Course Objectives:

1. To provide knowledge and skill to use tools, machines, equipment, and measuring instruments, which are used in manufacturing industries.
2. To educate students for Safe handling of machines and tools in manufacturing environment

Course Outcomes:

1. The student will be able to apply concept related to workshop safety & use of measuring instruments during process of manufacturing.
2. The student will be able suitably select basic manufacturing practices for making of component.
3. The students will be able to manufacture/produce given product from raw material using different manufacturing methods.

Unit – I: Introduction to Workshop Safety and Measuring Instruments: (05)

- Safety precautions while working in shop, safety equipment's and their use.
- Brief introduction to instruments like – Steel rule, Calipers, VernierCaliper, Micrometer, etc. Least counts, common errors and care while using them, use of marking gauge, 'V'block and surface plate.
- Introduction & working of different tools used in workshop.

Unit – II: Manufacturing Practice:(Any Two Trades) (13)

- Fitting: Preparation of joints, markings, cutting and filling for making joints like V or T for making part of any component.
- Carpentry: Wood working consists of planning, marking, sawing, chiseling and grooving to make joint like lap, T, dovetail.
- Tin smithy: Making of small parts using sheet metal such as Tray, Funnel.
- Welding Joints: Introduction to use of MIG/ TIG, arc welding for making joints like Lap, Butt joint.

Unit – III: Information technology: (06)

- Identify the peripherals of computer components in a CPU and its functions
- Disassemble and assemble the PC back to working condition
- Loading of operating system.

Unit – IV: Plumbing (06)

- Hands on practice on Cutting, bending and external threading of GI pipes using Die
- Plumbing on PVC pipes.
- Different Joint preparation on GI & PVC Pipes

Text Books:

1. Choudhary, Hajara**'Elements of Workshop Technology**', Media Promotors& Publishers, (1997).
2. Raghuvanshi B.S. "Workshop Technology" Vol. I &II,DhanpatRai& Sons, (1998).
3. H.S. Bawa'Workshop Technology' Vol.-I by, TMH Publications, New Delhi, (2009).
4. Gupta and Kaushik "Workshop Technology: Vol. – I by, New Heights,(1999).

Reference Books:

1. Chapman W.A. J and Arnold E. '**Workshop Technology-part I**' Viva low priced Student, (1998).

Autonomous Programme Structure of
Second Year B. Tech. Information Technology
Academic Year : 2019-2020

S. Y. B. Tech. Information Technology Semester – I										
Course Code	Course Title	Teaching Scheme Hours /Week			Examination Scheme				Marks	Credit
		Lecture	Tutorial	Practical	In Semester	End Semester	Oral	Practical		
IT 2101	Discrete Structures	3	1	0	50	50	0	0	100	4
IT 2102	Digital Systems	3	1	0	50	50	0	0	100	4
IT 2103	Data Structures I	3	0	0	50	50	0	0	100	3
IT 2104	Network Fundamentals	3	1	0	50	50	0	0	100	4
BSH 2101	Principles of Economics and Finance	3	0	0	50	50	0	0	100	3
IT 2105	Digital Systems Laboratory	0	0	2	0	0	0	25	25	1
IT 2106	Data Structures I Laboratory	0	0	4	0	0	0	50	50	2
IT 2107	Web Engineering Technology Laboratory	0	0	2	0	0	0	25	25	1
AC 2101	Self Expression	0	0	2	0	0	0	0	0	No Credit
	Total	**15**	**3**	**10**	**250**	**250**	**0**	**100**	**600**	**22**
	Grand Total	**28**			**600**				**600**	**22**

AC 2101 -- Audit Course: Self Expression

1. Dance
2. Drawing / Painting / Sketching
3. English Communication Skill
4. Film Appreciation
5. Origami
6. Theatre

MKSSS's Cummins College of Engineering for Women Karvenagar, Pune-411052

DEAN ACADEMICS
MKSSS's Cummins College of Engineering for Women Karvenagar, Pune-411052

Principal
MKSSS's Cummins College of Engg. For Women, Karvenagar, Pune-52.

APPROVED BY
Governing Body Members
MKSSS's Cummins College of Engineering for Women
Karvenagar, Pune-411052

IT 2101 Discrete Structures

Teaching Scheme:

Lectures: **3** Hrs/Week

Tutorial: **1** Hr/Week

Examination Scheme:

In-Semester: **50** Marks

End-Semester: **50** Marks

Credits: 4

Course Objectives:

1. Learn the concepts of propositions and propositional logic
2. Learn the concepts of sets operations and functions
3. Learn the fundamentals of counting, permutations and combinations
4. Learn the relations, its representations and properties
5. Learn the concepts of graph, its terminology, representation, connectivity, and its Applications.
6. Learn the concepts of tree, tree traversals and applications

Course Outcomes:

By the end of the course, students should be able to

1. Solve real world problem using sets and functions
2. Use proposition and propositional logic for drawing conclusions
3. Demonstrate the application of discrete structures using relations
4. Apply graphs as models to variety of domains
5. Apply trees in simple applications of computation
6. Evaluate the combinatorial problems

Unit – I: Sets and Functions **(07)**

Sets: Introduction to Power set, Cartesian products ; Set Operations: Introduction, Generalized union and intersection, Computer representation of sets; Functions: Introduction, One-to-One and Onto Functions, Inverse function and Composition of Functions

Unit – II: Propositional Logic **(06)**

Propositional Logic: Introduction, Proposition, Conditional Statements, Truth tables of compound proposition; Propositional equivalences: Introduction, Logical Equivalences, Constructing new logical equivalences; Preliminaries of predicates and quantifiers: Introduction, Predicates, Quantifiers, Negating quantified expressions

Unit – III: Relations **(08)**

Relations and Their Properties: Introduction, functions as relation, relations on set, Properties of relations, combining relations; n-ary Relations and Their Applications: Introduction, n-ary relations, operations on n-ary relations; Representing Relations:

Representing relations using matrices, Representing relations using digraph; Closures of Relations: Introduction, Closures, paths in directed graph, transitive closure, Warshall's algorithm; Equivalence Relations: Introduction, Equivalence relation, Equivalence classes and partition; Partial Orderings: Introduction, Hasse Diagrams, Maximal and Minimal elements, Lattices, discrete numeric functions

Unit – IV: Graphs **(06)**

Graphs and Graph Models , Graph Terminology and Special Types of graphs, Representing Graphs and Graph Isomorphism, Connectivity, Euler and Hamilton Paths , Shortest-Path Problems, Planar Graphs, Graph Coloring

Unit – V: Trees **(06)**

Introduction to Trees, Applications of Trees: Introduction, Binary search trees, Prefix codes, Tree Traversal: Preorder, in-order and post-order traversals , Minimum Spanning Trees: Introduction, Prim's algorithm, Kruskal's algorithm

Unit – VI: Counting **(07)**

The Basics of Counting: Introduction, Basic counting principles, Inclusion exclusion principle; The Pigeonhole Principle: Introduction, Generalized pigeonhole principle; Permutations and Combinations: Introduction, permutations, combinations; Binomial

Coefficients and Identities; Generalized Permutations and Combinations: permutation with repetition, combination with repetition; Generating Permutations and Combinations: Generating permutations, generating combinations

Text Books:

1. Kenneth H. Rosen, **"Discrete Mathematics and Its Applications",** Tata McGraw-Hill (7th Edition) (2012)

Reference Books:

1. C. L. Liu, "**Elements of Discrete Mathematics**", Tata McGraw-Hill (2^{nd} Edition)

IT 2102 Digital Systems

Teaching Scheme: **Examination Scheme:**

Lectures: **3** Hrs/Week In-Semester: **50** Marks

Tutorial: **1** Hr/Week End-Semester: **50** Marks

Credits: 4

Course Objectives:

1. To learn and understand basic digital design techniques
2. To develop, design and implement combinational and sequential logic circuits
3. To learn programmable logic devices
4. To introduce computer arithmetic

Course Outcomes:

On completion of the course, student will be able to Explain–

1. Apply knowledge of number systems, codes, Boolean algebra and use necessary A.C, D.C Loading characteristics as well as functioning while designing with logic gates.
2. Use logic function representation for simplification with K-Maps and analyze as well as design Combinational logic circuits using SSI & MSI chips.
3. Analyze Sequential circuits like Flip-Flops (Truth Table, Excitation table), their conversion & design the applications.
4. Identify the Digital Circuits, Input/Outputs to replace by FPGA

Unit – I: Number System and Logic Families (05)

Introduction to digital electronics & Boolean algebra. Number Systems - Binary, Octal, Hexadecimal and their conversions. Signed Binary number representation and Arithmetic's: Signed & True Magnitude, 1's complement, 2's complement representation and arithmetic's.
Codes: BCD, Excess-3, Gray code, Binary Code and their conversion. Switching characteristics of BJT & FET, IC Characteristics. TTL: Standard TTL characteristics, Operation of TTL NAND gate, Subfamilies, totem pole, CMOS: Standard CMOS characteristics, operation of CMOS NAND, Subfamilies, Comparison of TTL & CMOS, Interfacing: TTL to CMOS and CMOS to TTL

Unit – II: Combinational Logic Design (08)

Logic minimization: Representation of truth-table, Simplification of logical functions, Minimization of SOP and POS forms, don't care Conditions. Reduction techniques: K-Maps, Quine - McClusky technique. CLC design using SSI chips – Code converters, Adder, Subtracter, n bit Binary adder, look ahead carry generator. Magnitude comparator. Introduction to MSI functions & chips - Multiplexers, Decoder / Demultiplexer, Encoder, Binary adder. CLC design using MSI chips – BCD & Excess 3 adder & subtracter, Implementation of logic functions using MSI chips

Unit – III: Sequential Logic (06)

Introduction to sequential circuits. Difference between combinational circuits and sequential circuits, memory element – latch. Flip- Flops: Design, truth table, excitation table of SR, JK, D, T flip flops. Study of flip flops with asynchronous and synchronous Preset & Clear, Master Slave configuration, conversion from one type to another type of flip flop. Application of flip-flops – Bounce elimination switch, Counters- asynchronous, synchronous and modulo counters study of modulus n counter ICs & their applications to implement mod counters.

Unit – IV: Sequential Logic Design (08)

Registers- Buffer register, shift register types - SISO, SIPO, PISO & PIPO, applications of shift registers - ring counter, twisted ring counter, study of universal shift register. Sequence generators using counters & shift register, Pseudo Random Binary Sequence Generator. Basic design steps-State diagram, State table, State reduction, State assignment, Mealy and Moore machines representation, Implementation, finite state machine implementation, sequence detector using Moore & Mealy model.

Unit – V: Programmable Logic Devices (06)

Algorithmic State Machines- ASM notations, charts (e.g.- counters, washing machine, lift controller, vending machine), design using multiplexer controller method (e.g.- counters). Introduction to PLD's – ROM, PAL, PLA, Design of 4 variable SOP using PLDs, Basic architecture of SPLD and CPLD, Study of CPLD architecture XC9572, Basic architecture of FPGA, CPLD. Design flow

Unit – VI: Computer Arithmetic (07)

A Brief History of computers, Von Neumann Architecture, Harvard architecture, Bus Interconnection, Scalar Data Types, Fixed and Floating point numbers, Booths algorithm for multiplication and its Hardware Implementation, Division: Restoring and Non Restoring algorithms, IEEE standards of Floating point representations, Floating point arithmetic.

Text Books:

1. R.P. Jain, "Modern Digital Electronics", 3rd Edition, Tata McGraw-Hill, ISBN: 0–07–049492–4

Reference Books:

1. W. Stallings, "**Computer Organization and Architecture: Designing for Performance**", 8th Edition, Prentice Hall of India, 2010, ISBN 13: 978-0-13-607373-4
2. C. Hamacher, V. Zvonko, S. Zaky, "**Computer Organization**", 5th edition, McGraw Hill, 2002, ISBN: 007-120411-3

IT 2103 Data Structures I

Teaching Scheme:

Lectures: **3** Hrs/Week

Examination Scheme:

In-Semester: **50** Marks

End-Semester: **50** Marks

Credits: 3

Course Objectives:

1. To learn logic building using algorithm for problem solving
2. To learn logic building for puzzles and games
3. To learn use of different data structures and algorithm asymptotic notations.
4. To learn use of different searching and sorting techniques
5. To learn linear data structures using sequential organization and recursion concept.
6. To learn linear data structures using linked organization.

Course Outcomes:

1. Apply appropriate programming language constructs to develop logical steps to solve a given real world problem.
2. Select appropriate searching and/or sorting techniques for application development.
3. Analyze algorithm complexities and use appropriate algorithms to solve a given problem
4. Select appropriate sequential and linked organization of data structures to solve a given problem

Unit – I: Introduction to Algorithm and Logic building (06)

Concept of algorithm, Algorithmic thinking and Logic building, Solving specific real world problems such as in numerical methods, quantitative aptitude etc. using Operators, control structures, enumeration, structure, union, macros, arrays, functions and parameter passing, scope rules, string manipulation, matrix operations.

Unit – II: Logic building for Puzzles/ Games and File Organization (06)

Logic for password cracking (Brute Force – all possible permutations), puzzle solving & creation like Sudoku, magic square, eight queen, logical games like mine sweeper, connect dots, tic-tac-toe, debugging, dry-run, understand different codes
File Organization: file operations, keyword search

Unit – III: Introduction to Data structures and Analysis of Algorithms (06)

Introduction to Data Structures: Types of data structures, Abstract Data Types, Analysis of algorithm: frequency count and its importance in analysis of an algorithm, Time complexity & Space complexity of an algorithm, Best, Worst and Average case analysis of algorithm.

Unit – IV: Searching and sorting techniques (08)

Need of searching and sorting, Concept of internal and external sorting, sort stability. Searching methods: Linear and binary search algorithms their comparison and complexity analysis
Sorting methods: Bubble, selection, insertion, merge, quick, bucket sort and their complexity analysis.

Unit – V: Logic building using linear data structures and recursion (06)

Concept of Linear data structures, ordered list, Multidimensional arrays and their storage representation. Sparse matrix using arrays - addition, polynomial representation.
Concept of recursion and logic building using iterative and recursive methods, Recursive algorithms e.g. Factorial, Fibonacci series, etc. Use of implicit stack in recursion

Unit – VI: Linked List (08)

Concept of linked organization, singly linked list, doubly linked list, circular linked list. Linked list as an ADT. Representation and manipulations of polynomials using linked lists, comparison of a sequential and linked memory organization, concept of Generalized Linked List, representation polynomial using GLL.

Text Books:

1. R. Gilberg, B. Forouzan, "**Data Structures: A pseudo code approach with C"**, Cenage Learning

Reference Books:

1. Dennis Ritchie, Kernighan, "**The C Programming Language'**, Prentice Hall
2. E. Horowitz, S. Sahani, S. Anderson-Freed, "**Fundamentals of Data Structures in C"**, Universities Press
3. G. A. V. Pai, "**Data structures and Algorithms"**, McGraw Hill
4. Jon Bentley, **"Programming Pearls"**, Addison Wesley

IT 2104 Network Fundamentals

Teaching Scheme:

Lectures: **3** Hrs/Week

Tutorial: **1** Hr/Week

Examination Scheme:

In-Semester: **50** Marks

End-Semester: **50** Marks

Credits: 4

Course Objectives:

1. To understand fundamentals of communication systems.
2. To acquaint themselves with layered model used in computer networks.
3. To understand OSI and TCP/IP models.
4. To understand analyse MAC layer protocols and LAN technologies.

Course Outcomes:

1. Enumerate the layers of the OSI model and TCP/IP.
2. To differentiate between media access schemes.
3. Design the IP addressing schemes for a computer network.
4. Familiarity with the basic protocols of computer networks, and how they can be used to assist in network design and implementation.

Unit – I: Network – Centric World **(6)**

Communicating in a Network-Centric World, The Architecture of the Internet, Trends in networking, LAN, WAN, MAN, Networking Devices, Network Topologies Point to Point, Point to Multipoint Topologies.

Unit – II: Communicating over the Network **(6)**

The platform for communications, Protocols, OSI Model, TCP/IP Model, Protocol Data Units and Encapsulation, Comparison between OSI and TCP/IP Model, Network Addressing.

Unit – III: Network Layer **(8)**

IP Addressing, Communication from Host to Host ,Network Layer Protocol, Packaging the Transport Layer PDU ,IPv4 Packet Header, Subnetting, Static Routing ,Dynamic Routing ,Routing Protocols

Unit – IV: Ethernet **(6)**

Ethernet Basics, Collision Domain , Broadcast Domain, CSMA/CD , Half- and Full-Duplex Ethernet, Ethernet at the Data Link Layer, Ethernet Addressing , Ethernet Frames ,Channel Bonding, Ethernet at the Physical Layer.

Unit – V: Physical Layer **(7)**

The Theoretical Basis for data communication, Digital Modulation and Multiplexing, The Public Switched Telephone Network and Cable Television, Community Antenna Television, Internet over Cable, Spectrum Allocation, Cable Modems, ADSL Versus Cable, Network Interface.

Unit – VI: Data Link Layer **(7)**

Data Link Layer Design Issues, Error Detection and Error Correction, Sliding Window Protocol, Medium Access Control Sub layer, Channel Allocation Problem, Ethernet MULTIPLE ACCESS PROTOCOLS, ALOHA, Carrier Sense Multiple Access Protocols, Collision-Free Protocols,

Limited-Contention Protocols, Wireless LAN Protocols,

Text Books:

1. Mark A. Dye, Rick McDonald, Antoon W. Rufi, **"Network Fundamentals",** Cisco Press (2008)

Reference Books:

1. Andrew S. Tennabaum, David J. Weatherall **"Computer Networks",** Pearson (5^{th}edition), (2011)

IT 2105 Digital Systems Laboratory

Teaching Scheme: **Examination Scheme:**

Practical : **2** Hrs/Week Practical: **25** Marks

Credits: 1

Course Objectives:

1. To learn and understand basic digital design techniques.
2. To develop design and implementation skills of combinational and sequential logic circuits.
3. To introduce computer Arithmetic

Course Outcomes:

On completion of the course, student will be able to explain–

1. Apply knowledge of number systems, codes, Boolean algebra and use necessary A.C, D.C Loading characteristics as well as functioning while designing with logic gates.
2. Use logic function representation for simplification with K-Maps and analyze as well as design Combinational logic circuits using SSI & MSI chips.
3. Analyze Sequential circuits like Flip-Flops (Truth Table, Excitation table), their conversion & design the applications.
4. Identify the Digital Circuits, Input/Outputs to replace by FPGA, Advanced processor organization

Suggested List of Laboratory Assignments

1. Design (truth table, K-map) and implementation of 4-bit BCD to Excess-3 and Excess-3 to BCD Code converters.
2. Design (truth table, K-map) and implementation of 4 bit BCD & Excess 3 Adder using IC7483.
3. Implementation of logic functions using multiplexer IC 74153 & decoder IC 74138. (Verification, cascading & logic function implementation)
4. Design (State diagram, state table & K map) and implementation of 3 bit Up and Down Asynchronous Counter using master slave JK flip-flop IC 7476
5. Design (State diagram, state table & K map) and implementation of 3 bit Up and Down Synchronous Counter using master slave JK flip-flop IC 7476
6. Design and implementation of Module ‘n’ counter with IC7490 and IC 74191
7. Design (State Diagram, State Table, K Map) and implementation of Sequence Generator using Shift Register IC 74194.
8. Design and implement unsigned binary multiplication (3 bit)

Text Books:

1. R.P. Jain, “**Modern Digital Electronics**”, 3rd Edition, Tata McGraw-Hill, ISBN: 0–07–049492–4

Reference Books:

1. W. Stallings, “**Computer Organization and Architecture: Designing for Performance**”, 8th Edition, Prentice Hall of India, 2010, ISBN 13: 978-0-13-607373-4

2. C. Hamacher, V. Zvonko, S. Zaky, "**Computer Organization**", 5th edition, McGraw Hill, 2002, ISBN: 007-120411-3

IT 2106 – Data Structures I Laboratory

Teaching Scheme: **Examination Scheme:**

Practical : **4** Hrs/Week Practical: **50** Marks

Credits: **2**

Course Objectives:

1. To learn Python constructs
2. To learn algorithm development and analysis of algorithms
3. To learn linear data structures and their applications
4. To learn different searching and sorting techniques
5. To build logic to solve real world problems
6. To learn debugging to understand different codes & detect logical errors

Course Outcomes:

On completion of the course, student will be able to –

1. Implement appropriate searching and/or sorting techniques to solve a given problem
2. Implement algorithms to illustrate use of data structures such as array, linked list
3. Implement algorithms to create and manipulate database using sequential file organization
4. Debug different code snippets

Suggested List of Laboratory Assignments (13 assignments)

Group A Assignments (Python programming) (Any 5)

1. To check whether a given input number is prime or not
2. To develop a password cracker (brute force - permutations)
3. To develop tic-tac-toe game
4. a) Sort the set of strings in ascending order using Bubble sort and descending order by using Selection sort or Insertion sort. (Display pass by pass output) b) Search a particular string using binary search with and without recursion.
5. Implement Quick Sort to sort the given list of numbers. Display corresponding list in each pass. (with and without recursion)
6. Implement a doubly linked list with following options
   a) Insertion of a node at any location b) Deletion of a node from any location c) Display a linked list d) Display in linked list in reverse e) Reverse the linked list without using additional data structure, no data swapping

Group B Assignments (C programming) (Compulsory)

1. There are 10 students in art class and 15 students in dance class. 8 students are enrolled in both activities. (Sets).
   a) Find the students who are enrolled in both the activities
   b) Find the students who are enrolled only in art class
   c) Find all the student without repetition
2. Create a Database for employee salary calculation using array of structures and perform following operations on it:
   a) Create Database b) Display Database (tabular format) c) Add a record d) Search a record e) Modify a record f) Delete a record g) Search can be in different manner e.g. Search all records having percentage more than 70.
3. Implement sequential file and perform following operations:
   a) Display b) Add records c) Search record d) Modify record e) Delete record
4. Implement a singly linked list with following options
   a) Insertion of a node at any location
   b) Deletion of a node from any location
   c) Display a linked list
   d) Display in linked list in reverse
   e) Reverse the linked list without using additional data structure, no data swapping
5. Implement a doubly linked list with following options
   a) Insertion of a node at any location
   b) Deletion of a node from any location
   c) Display a linked list
   d) Display in linked list in reverse
   e) Reverse the linked list without using additional data structure, no data swapping

Group C Assignments (C programming) (Any 2)

1. Solve Simultaneous Equations in Three Variables (Matrix)
2. Implement following operations on string with / without pointers (without using library functions)
   a) Length b) Copy c) Reverse d) String comparison e) Palindrome f) Substring g) Search and replace character h) Password validation i) Code / decode
3. Implement polynomial using CLL and perform
   a) Addition of Polynomials b) Multiplication of polynomials c) Evaluation of polynomial

   Implement Generalized Linked List to create and display operations

Group D Assignment (Any programming language) (Any 1)

One assignment to be carried out by a group of 2 students.
Each group will design an application using the appropriate data structures – to solve real world problem (proof of concept). The group requires getting the application approved by the respective faculty member.

1. Unit / number system conversions
2. Verification of amount in digits and in words (e.g. as given on cheque)
3. Result analysis of class data (e.g. no. of first classes etc.)
4. Implementation of skip list
5. Operations on polynomials (e.g. add, multiply, evaluate)
6. Searching & counting no. of occurrence & location (line no) of a word in a given
7. Searching & counting no. of occurrence & location (line no) of a word in a given text file

8. Implementation of numerical methods (e.g. Runge Kutta)
9. Implementation to generalize Time / Train based aptitude question (R. S. Agarwal)
10. Recursive solution to problems (e.g. Tower of Hanoi)
11. Develop games (e.g. Tic-tac-toe, sudoku)
12. Text editor (Hint – GLL)
13. Code beautifier (e.g. int a ; - keyword shown in blue color, others in black)

Text Books:

1. Steve McConnell, "**Code complete**", Second edition, 2nd ed. Redmond, WA: Microsoft Press, 2007.
2. E. Balagurusamy, "**Introduction to Computing and Problem Solving Using Python**", McGraw Hill, ISBN : 9352602587

Reference Books:

1. E. Horowitz, S. Sahani, S. Anderson-Freed, "**Fundamentals of Data Structures in C**", Universities Press, 2008
2. R. Gilberg, B. Forouzan, "**Data Structures: A pseudo code approach with C**", Cenage Learning, ISBN 9788131503140.
3. Yashwant Kanetkar, "**Pointers in C**", BPB Publication
4. Rance Necaise, "**Data Structures and Algorithms Using Python**", Wiley, ISBN : 9788126562169

IT 2107 Web Engineering Technology Laboratory

Teaching Scheme: **Examination Scheme:**

Practical: **2** Hrs/Week Practical: **25** Marks

Credits: **1**

Course Objectives:

1. To understand various application layer protocols for its implementation in client/server environment

Course Outcomes:

By the end of the course, students should be able to

1. Apply basics of web designing
2. Design a simple web application
3. Implement dynamic web pages
4. Establish client and server-side communication

Group A

A. HTML

Create a registration form using HTML form input elements viz. textbox, text area, radio button and drop down menu, check box, submit, file and reset button. Field should contain name, address, birth-date, qualification, email, phone number, gender, comments, attach photo etc. Use HTML Form elements wherever required. Align all elements using table.

B. CSS

Create a horizontal navigation bar in DIV using external CSS which contain home, about, gallery, enquiry, contacts menus. Also create the same bar in vertical alignment in another DIV in same page.

C. Java Script

1. Write a Java script to create a simple calculator.
2. Write a Java script that read ten numbers and display the count of negative and positive numbers and count of zero from the list.
3. Create form validation program that checks the empty values from that form and alert back using alert function. Use at least 5 components.

D. PHP

1. Create a PHP program in which two values submitted using form and calculate its addition, subtraction, multiplication, modulation, average and division on the same page. Find the greatest number between them and square of each of them using PHP function.
2. Write PHP script to display the squares and cubes of 1 to 10 numbers in tabular format.
3. Write PHP script to validate Email address.
4. Create a login form using session handling in PHP. After successful login display name, address and other details in tabular format of logged user.

E. XML

Write an XML schema that provides tabulated information related to expected height (in cms) and weight (in kgs) for male and female separately for the age groups starting with 5-10 years, 15-20 years, and so on.

Group B

Design and develop web site in group of 2 using above all learnt technology.

Text Books:

1. **'Web Technologies Black Book: HTML, JavaScript, PHP, Java, JSP, XML and AJAX'** by Kogent Learning Solutions Inc.

Reference Books:

1. Steven M. Schafer, '**HTML, XHTML and CSS'**, Fourth Edition, Wiley India Edition. ISBN: 978-81-265-1635-3.

S. Y. B. Tech. Information Technology Semester – II										
Course Code	Course Title	Teaching Scheme Hours /Week			Examination Scheme				Marks	Credit
		Lecture	Tutorial	Practical	In Semester	End – Semester	Oral	Practical		
IT 2201	Data Structures II	3	0	0	50	50	0	0	100	3
IT 2202	Computer Network	3	1	0	50	50	0	0	100	4
IT 2203	Computer Organization and Architecture	3	1	0	50	50	0	0	100	4
IT 2204	Object Oriented Paradigms	3	1	0	50	50	0	0	100	4
BSIT 2201	Engineering Mathematics III	3	1	0	50	50	0	0	100	4
IT 2205	Data Structures II Laboratory	0	0	4	0	0	0	50	50	2
IT 2206	Network Laboratory	0	0	2	0	0	0	25	25	1
IT 2207	Computer Organization and Architecture Laboratory	0	0	2	0	0	0	25	25	1
IT 2208	Object Oriented Programming Laboratory	0	0	2	0	0	0	25	25	1
	Total	15	4	10	250	250	0	125	625	24
	Grand Total	29			625				625	24

DEAN ACADEMICS
MKSSS's Cummins College of Engineering for Women Karvenagar, Pune-411052

Principal
MKSSS's Cummins College of Engg For Women, Karvenagar, Pune-52

APPROVED BY
Governing Body Members
MKSSS's Cummins College of Engineering for Women Karvenagar, Pune-411052

MKSSS's CUMMINS COLLEGE OF ENGG FOR WOMEN KARVENAGAR PUNE-52.

IT 2201 Data Structures II

Teaching Scheme:
Lectures: **3** Hrs/Week

Examination Scheme:
In-Semester: **50** Marks
End-Semester: **50** Marks
Credits: **3**

Course Objectives:

1. To learn concepts and use of stack and queue data structures.
2. To learn basic tree data structure and traversals with BST
3. To learn graphs, traversals and algorithms on graph data structure.
4. To learn symbol tables and hashing with their applications.
5. To study some advanced tree concepts.
6. To learn different file organizations and their use in practice.

Course Outcomes:

1. Select appropriate data structure to solve real-world problem.
2. Solve problem involving linear data structures.
3. Solve problem involving nonlinear data structures.
4. Make use of different hashing techniques and compare their performances.

Unit – I: Stacks and Queues **(07)**

Concept of stack, stack as ADT, Implementation of stack using array and linked organization, multistacks, use of stack- Recursion, expression conversion & evaluation.
Concept of queues as ADT, Implementation using array and linked organization. multiqueues, priority queue.

Unit – II: Trees **(07)**

Difference in linear and non-linear data structure, Trees and binary trees-concept and terminology. Expression tree. Conversion of general tree to binary tree. Binary tree as an ADT. Recursive and non-recursive algorithms for binary tree traversals, Binary search trees, Binary search tree as ADT

Unit – III: Graphs **(07)**

Graph as an ADT, Representation of graphs using adjacency matrix and adjacency list, Depth First Search and Breadth First Search traversal. Prim's and Kruskal's algorithms for minimum spanning tree, shortest path using Warshall's and Dijkstra's algorithm.

Unit – IV: Tables **(07)**

Symbol Table: Symbol Table, Huffman's algorithm, Heap data structure, applications of heap, Heap sort **Hash table:** hashing function, collision resolution techniques- linear probing, rehashing, chaining without replacement and chaining with replacement.

Unit – V: Advance Trees **(06)**

Concept of threaded binary tree. Preorder and In-order traversals of in-order threaded binary tree, AVL Trees, OBST

Unit – VI: File organization (06)

External storage devices, File, File types and file organization (sequential, index sequential and Direct access), Primitive operations and implementations for each type and comparison

Text Books:

1. R. Gilberg, B. Forouzan, "**Data Structures: A pseudo code approach with C"**, Cenage Learning

Reference Books:

1. Bruno R Preiss, "**Data Structures and Algorithms with object-oriented design patterns in C++"**, Wiley India Edition
2. E. Horowitz, S. Sahani, S. Anderson-Freed, "**Fundamentals of Data Structures in C"**, Universities Press
3. G. A.V. Pai , "**Data structures and Algorithms"**, McGraw Hill
4. Y. Langsam, M. Augenstin, A. Tannenbaum, "**Data Structures using C and C++"**, Prentice Hall of India,

IT 2202 Computer Networks

Teaching Scheme:

Lectures: **3** Hrs/Week — In-Semester: **50** Marks

Tutorial: **1** Hr/Week — End-Semester: **50** Marks

Credits: **4**

Course Objectives:

1. To understand routing and network layer.
2. Understanding of TCP and UDP key functions.
3. Understanding the role of transport layer in congestion control, fairness and stability of Internet.
4. To understand Wireless Technologies.

Course Outcomes:

1. Analyse the usage of various protocols at the network layer.
2. Compare the routing algorithms.
3. Analyse the usage of various protocols at the transport layer.
4. Comprehend the wireless transmission media.

Unit – I: Internetworking **(6)**

Internetworking Basics, OSI Model, Data Encapsulation, Introduction to TCP/IP, the process/Application layer Protocols, The Host-to Host Layer Protocols. The Internet Layer Protocols, Internet Protocol.

Unit – II: Introduction to Routing and Packet Forwarding **(6)**

Inside the Router, CPU, NVRAM, Router Interfaces, Routers and the Network Layer , Command Line Interface Configuration and Addressing , Basic Router Configuration , Building the Routing Table, IP Routing, Path Determination and Switching Functions Protocols.

Unit – III: IP Routing **(7)**

Routing Basics, Distance vector routing Protocols, Link state routing protocols. Routing Information protocol, Enhanced Interior Gateway Routing Protocol, Open Shortest Path First. Virtual Local area Networks, Network address translation.

Unit – IV: Transport Layer **(7)**

Transport layer duties and functionalities, application expectations and IP delivery semantics.
UDP: UDP functionality, UDP Header.
TCP: TCP Features, byte-stream, Connection-oriented, TCP Header Format, 2-way, 3-way Handshake, TCP State Diagram, TCP Sliding Window, Congestion Control Algorithms: Leaky Bucket, Token Bucket, Congestion Avoidance. UNIX Sockets. M/M/1 queue analysis.

Unit – V: Application Layer **(6)**

Client/Server Model, Telnet, Domain Name System, File Transfer protocol: FTP, TFTP, Hyper Text Transfer Protocol, POP3, IMAP, SMTP, E-mail, MIME, Simple Network Management Protocol

Unit – VI: Wireless Technologies & SDN **(6)**

Introduction to wireless internetwork, IEEE 802.11, Cellular Technology, WLAN, Internet of Things, Bring Your Own Device. Software defined networking, concept, architecture, applications.

Text Books:

1. Andrew S. Tennabaum, David J. Weatherall **'Computer Networks',** Pearson (5th edition), (2011)
2. Behrouz Forouzan ,**'TCP/IP Protocol Suite',** Mc-Graw Hill, (4th Edition) (2010)

Reference Books:

1. Theodore S. Rappaport, **'Wireless Communications'**, Prentice Hall (2nd Edition) (2002)
2. Rick Graziani, Allan Johnson, **'Routing Protocols and Concepts ,** Cisco Press (2011)

IT 2203 Computer Organization and Architecture

Teaching Scheme:
Lectures: **3** Hrs/Week
Tutorial: **1** Hr/Week

Examination Scheme:
In-Semester: **50** Marks
End-Semester: **50** Marks
Credits: 4

Course Objectives:

1. To understand configuration of Computer Systems
2. To understand fundamental working of Computer Systems.
3. To study architecture and features of 8086 microprocessor
4. To learn interfacing of 8086 microprocessor with I/O peripherals

Course Outcomes:

On completion of the course, student will be able to explain–

1. Structure and function of Computer System
2. Architectural details of 8086 microprocessor
3. Memory management and Interrupts of 8086
4. Interfacing of microprocessor with I/O peripherals

Unit – I: Basic Processing Unit and Machine Instructions (07)

Fundamental Concept of basic processing Unit: Register Transfer, Arithmetic Logic Operation, Fetching and storing a word, Execution of Complete Instruction. Instruction and Instruction Sequencing: Instruction Types, Straight line Sequencing, branching, Condition codes. Addressing Modes

Unit – II: Processing Unit 8086 Microprocessor: Architecture, (08)

Instruction Descriptions and Assembler Directives

Introduction to 8086: internal Architecture, generation of physical address, minimum/maximum mode, study of 8086 supporting chips 8288(Latch), 8284(Clock Generator),8286(trans receiver),8288(Bus controller), Timing diagram read Write machine cycle. Introduction to assembly language programming- Instruction Descriptions, Assembler Directives.

Unit – III: Assembly Language Programming and Interrupt structure (07)

Address translation, addressing modes, Examples of programming, Procedures and Macros Interrupt Structure, Interrupt service routine, Interrupt vector table, hardware and software interrupts, INTR, NMI, Interrupt response, Execution of ISR, Priorities of interrupt

Unit – IV: Interfacing with 8086-I (07)

8259(Programmable Interrupt Controller)- Block Diagram, control and status register, Interfacing and programming. 8255(Programmable peripheral interface)- Block diagram control word, Interfacing ADC and DAC.

Unit – V: Interfacing with 8086-II (06)

8253/54(programmable interval timer/counter)- Block Diagram, control word. Modes of timer 8251(USART)- Features, Block Diagram, Control and Status register, operating modes.

Unit – VI: Parallel Organization **(05)**

Parallel Organization – Multiprocessors, Multicores & Clusters. Flynn's Taxonomy for Multiple Processor Organizations, Closely and Loosely Coupled Multiprocessors Systems, Symmetric Multiprocessor (SMP) Organization, Multi-threading – Fine Grained, Coarse Grained & Simultaneous (SMT) Threading, Chip Multiprocessing, Cluster Configuration, UMA, NUMA & CC-NUMA. Multicore Architectures – Hardware & Software Issues in Multicore Organization, Multicore Organizations, Intel X86 Multicore Organizations – Core Duo & Core i7.

Text Books:

1. C. Hamacher, V. Zvonko, S. Zaky, "**Computer Organization**", 5th edition, McGraw Hill, 2002, ISBN: 007-120411-3
2. Douglas Hall, "**Microprocessors and Interfacing, Programming and Hardware**", McGraw-Hill, ISBN: 0–07–100462–9

Reference Books:

1. W. Stallings, "**Computer Organization and Architecture: Designing for Performance**", 8th Edition, Prentice Hall of India, 2010, ISBN 13: 978-0-13-607373-4

IT 2204 Object Oriented Paradigms

Teaching Scheme:

Lectures: **3** Hrs/Week

Tutorial: **1** Hr/Week

Examination Scheme:

In-Semester: **50** Marks

End-Semester: **50** Marks

Credits: 4

Course Objectives:

1. The students should be able to understand abstraction
2. The students should be able to understand the encapsulation
3. The students should be able to understand the inheritance and polymorphism.
4. The students will be able to convert a small problem description in an object oriented code

Course Outcomes:

1. The students will be able to abstract required properties and behavior of a class from a description.
2. The students will be able to apply inheritance to a given problem description.
3. The students will be able to derive encapsulation and polymorphic behavior from a given problem description.
4. The students will be able to determine all applicable object oriented features from the given description.

Unit – I: Building blocks of Object Oriented Programming (06)

Revision of procedural programming, Limitations of procedural programming, Algorithmic decomposition Vs Object Oriented decomposition.

Concepts of Class, Object, State of an Object, behavior of an object and identity for an object.

Introduction to scope: private/ protected/ public/package level

Concepts of Information hiding, Abstraction and Encapsulation as what are those and their necessity.

Unit – II: Abstraction (06)

Writing a class with private instance variables and instance methods in appropriate scope, properties with accessor (getXXX) and modifier (setXXX) methods, and constructors. Effective use of comments such as class level, method level, and inline

Class as a user defined data type against primitive data types. Instantiating an object, using it through its abstraction. Introduction to terms ‘Reference’

Unit – III: Inheritance and substitution (06)

Method overloading, overloaded constructors, chaining of constructors. ‘this’ keyword and its concept. division into parts, composition, layers of specialization, subclass, subtypes, forms of inheritance, variations on inheritance, benefits and cost of inheritance

Best practices: naming conventions, packaging (name space).

Methods from Object class: rules for overriding equals(), hashCode() and toString().

Unit – IV: Polymorphism and code reuse (06)

Containment: Code reuse through containment of objects. Object as a smallest reusable unit. Distribution of responsibilities across application. Localization of impact due to changes in requirement.

Inheritance: Concept referring to generalization-specialization, inheritance for members according to the scope, code reuse, method overriding, polymorphism, effects of using base class reference for child class object, chaining of constructors (passing data to super class).

Unit – V: Abstract class and aggregation in Object orientation (06)

Abstract class, abstract methods, concept of Interface, final class/ method

Array of 'primitive data type' and Array of 'user defined data type', introduction to multi dimension array.

Unit – VI: Introduction to I/O Programming and Exception (06)

Introduction to language specific Collections framework, introduction to concept of List/ Set/ Map and techniques to iterate over them.

Text Books:

1. Kathy Sierra, **'OCA / OCP Java SE 7 Programmer I & II Study Guide, Chapter 1, 2 and 7** Oracle press (2014)

Reference Books:

1. Khalid A Mughal, **'A programmer's guide to Java SE 8 oracle certified associate'** Oracle press (2017)

BSIT 2201 Engineering Mathematics III

Teaching Scheme:
Lectures: **3** Hrs/Week
Tutorial: **1** Hr/Week

Examination Scheme:
In-Semester: **50** Marks
End-Semester: **50** Marks
Credits: **4**

Prerequisites:

1. Permutation and Combination
2. Complex numbers - Properties, Argand Diagram, Basic properties of integration.
3. Partial Fractions, Basic properties of integration, Beta and Gamma Functions.
4. Number System.
5. First order linear ordinary differential equations.

Course Objectives:

1. To recall and remember basics of Statistics, complex analysis, Fourier and Z-transforms, Number theory and Higher order LDE.
2. To understand the concepts of basic mathematical methods for solving Statistics, complex analysis, Fourier and Z-transforms, Number theory and Higher order LDE.
3. To apply these methods to solve engineering problems.
4. To analyze engineering problems and evaluate.

Course Outcomes:

On completion of the course, learner will be able to –

1. Represent and statistically analyze the given data. Compute probabilities of various events using conditional probability, probability distributions.
2. Obtain Fourier and Z Transforms of various functions. Solve integral equations using Fourier transforms and difference equations using Z transforms.
3. Analyse and apply concepts of analyticity for functions of complex variables.
4. Analyze and interpret the concepts of divisibility, congruence, greatest common divisor.
5. Formulate and solve higher order Linear Differential Equations, simple electrical circuits.

Unit – I: Statistics **(06)**

Measures of central tendency, Standard deviation, Coefficient of Variation, Moments, Skewness & Kurtosis, Probability, Theorems on probability, Conditional Probability & Bayes' theorem.

Unit – II: Probability Distributions **(06)**

Random Variables – Discrete & continuous, Mathematical expectations, Probability density functions, Standard Distributions – Binomial, Poisson, Normal, Lognormal.

Unit – III: Complex Analysis **(08)**

Functions of Complex variables, Analytic Functions, Cauchy Riemann-Equations, Cauchy's Integral Theorem, Cauchy's Integral Formula, Laurent's series, Residue theorem, Conformal mapping, Bilinear Transformation

Unit – IV: Transforms **(08)**

Z Transforms - Definition, standard properties, Z- Transform of standard sequences and their inverses, solution of difference equation.

Fourier Transforms - Complex exponential form of Fourier series, Fourier integral

theorem, sine and cosine integrals, Fourier transform, Fourier Sine and Cosine transform and their inverses.

(06)

Unit – V: Number Theory

Modular Arithmetic, Greatest common divisor, Euclid's algorithm, Chinese remainder theorem,
Fermat's theorem, Discrete Logarithm.

Unit – VI: Higher Order Linear Differential equation and application **(08)**

Higher order Linear differential equation with constant coefficients, Method of Variation of parameters, Cauchy's and Legendre's DE, Simultaneous DE, Modelling of electrical circuits.

Text Books:

1. B.S. Grewal, **'Higher engineering Mathematics'**, Khanna publishers, Delhi (40^{th} edition), (2008)
2. B. V. Ramana, **'Higher Engineering Mathematics'**, Tata McGraw Hill Publications (2007)
3. S.C. Gupta, V. K. Kapoor, '**Fundamental of Mathematical Statistics',** S. Chand & Sons (10^{th} revised edition). 2002
4. David M. Burton, **'Elementary Number Theory',** Tata McGraw Hill Publications (2012). 7^{th} edition.

Reference Books:

1. Peter V. O'neil,'**Advanced Engineering Mathematics',** Thomson Brooks / Cole,Singapore (5th edition) (2007).
2. Erwin Kreyszig,'**Advanced Engineering Mathematics'** Wiley Eastern Ltd. (8th Student Edition), (2004).
3. C.R.Wylie, L.C. Barrette, '**Advanced Engineering Mathematics'**, McGraw Hill Publications, New Delhi.(6^{th} edition)(2003)

IT 2205 – Data Structures II Laboratory

Teaching Scheme: **Examination Scheme:**

Practical: **4** Hrs/Week Practical: **50** Marks

Credits: **02**

Prerequisites:

IT 2106: Data Structures I Laboratory

Course Objectives:

1. To use linear data structures – stack & queue.
2. To learn non-linear data structures and their applications.
3. To learn different file organizations
4. To learn different hashing techniques
5. To understand use of data structures using OOP language

Course Outcomes:

Students will be able to

1. Implement algorithm to illustrate use of linear data structures such as stack, queue.
2. Implement algorithms to create/represent and traverse non-linear data structures such as trees, graphs.
3. Implement algorithms to create and manipulate database using different file organizations.
4. Implement and analyze different hashing techniques with respect to time and space complexity.

Suggested List of Laboratory Assignments (11 assignments)

Group A Assignments (C Programming)

1. Implement stack as an abstract data type using linked list and use this ADT for conversion of infix expression to postfix, prefix and evaluation of postfix expression.
2. Construct an expression tree from postfix/prefix expression and perform recursive and non-recursive In-order, pre-order and post-order traversals.
3. Implement binary search tree and perform following operations: a) Insert b) Delete c) Search d) Display e) Mirror image f) Display level-wise
4. Represent any real world graph using adjacency list /adjacency matrix find minimum spanning tree using Kruskal's algorithm.
5. Represent a given graph using adjacency matrix /adjacency list and find the shortest path using Dijkstra's algorithm (single source all destination).
6. Implement direct access file using hashing (chaining without replacement) perform following operations on it a) Create Database b) Display Database c) Add a record d) Search a record e) Modify a record

Group B: (Using Python programming) (Any2)

1. Implement priority queue as ADT using single linked list for servicing patients in an

hospital with priorities as a) Serious (top priority) b) medium illness (medium priority) c) General (Least priority).

2. Create Binary tree and perform following operations: a) Insert b) Display c) Depth of a tree d) Display leaf-nodes e) Create a copy of a tree
3. Consider a friends' network on face book social web site. Model it as a graph to represent each node as a user and a link to represent the friend relationship between them. Store data such as date of birth, number of comments for each user.
   a) Find who is having maximum friends b) Find who has post maximum and minimum comments c) Find users having birthday in this month. Hint: (Use adjacency list representation and perform DFS and BFS traversals)
4. Store data of students with telephone no and name in the structure using hashing function for telephone number and implement chaining with and without replacement.
5. A business house has several offices in different countries; they want to lease phone lines to connect them with each other and the phone company charges different rent to connect different pairs of cities. Business house want to connect all its offices with a minimum total cost. Solve the problem by suggesting appropriate data structures

Group C Assignments (C++ / Java) (Any 2)

1. Expression conversion using STL
2. Expression conversion using linked list
3. Binary Tree operations
4. Huffman coding
5. Sequential file handling

Group D Assignment (Any Programming Language) (Any 1)

One assignment to be carried out by a group of 2 students.

Each group will design an application using the appropriate data structures – to solve real world problem (proof of concept). The group requires to get the application approved by the respective faculty member.

1. Implementation of Tower of Hanoi (Non recursive implementation)
2. Recursive solution to problems (e.g. Tower of Hanoi)
3. Text editor (Hint – GLL)
4. Implementation of Process scheduling (e.g. long-term, short-term scheduler)
5. Implementation of AVL trees
6. Implementation of Loss less compression technique (Huffman) – encode & decode
7. Threaded binary tree – thread creation, display
8. Implementation of Hierarchical structure of organization (e.g. no. of first classes etc)
9. Simulation of college network
10. Searching & counting no. of occurrence & location (line no) of a word in a given text file
11. Formation of Magic square
12. Implementation to generalize Time / Train based aptitude question (R. S. Agarwal)
13. Develop games (e.g. Tic-tac-toe, sudoku)
14. Code beautifier (e.g. int a ; - keyword shown in blue color, others in black)

Text Books:

1. Steve McConnell, '**Code complete**', Second edition, 2nd ed. Redmond, WA: Microsoft

Press, 2007.
2. E. Horowitz, S. Sahani, S. Anderson-Freed, '**Fundamentals of Data Structures in C**', Universities Press, 2008

Reference Books:

1. R. Gilberg, B. Forouzan, '**Data Structures: A pseudo code approach with C**', Cenage Learning, ISBN 9788131503140.
2. Yashwant Kanetkar, '**Pointers in C**', BPB Publication
3. E. Balagurusamy, '**Introduction to Computing and Problem Solving Using Python**', McGraw Hill, ISBN: 9352602587
4. Rance Necaise, '**Data Structures and Algorithms Using Python**', Wiley, ISBN : 9788126562169

IT 2206 Network Laboratory

Teaching Scheme: **Examination Scheme:**

Practical: **2** Hrs/Week Practical: **25** Marks

Credits: **1**

Course Objectives:

1. To understand Routing and its Concepts.
2. To acquaint students with IP routing.
3. To understand dynamic Routing Protocols.
4. To understand Wireless Technologies.

Course Outcomes:

1. Apply routing protocols to a computer network.
2. Compare routing protocols by applying them to a computer network.
3. Apply application protocols to a computer network.
4. Calculate QoS parametrs of a computer network.

Suggested List of Laboratory Assignments

1. Configuration of Local Area Network.
2. Configuration of Static Routes on Router.
3. Configuration of Dynamic Routing Algorithm.
4. Implementation of Virtual LAN.
5. Configuration of EIGRP Protocol.
6. Configuration of OSPF Protocol.
7. Configuration of FTP, TELNET and DHCP.
8. Configuration of wireless network.

Text Books:

1. Antoon Rufi, Priscilla Oppenheimer, Belle Woodward, Gerlinde Brady, **'Network Fundamentals, CCNA Exploration Labs and Study Guide,** Pearson (2008)

Reference Books:

1. Andrew S. Tennabaum, David J. Weatherall **'Computer Networks',** *Pearson* (5^{th}edition), (2011)
2. Behrouz Forouzan ,**'TCP/IP Protocol Suite',** Mc-Graw Hill, (4^{th} Edition) (2010)

IT 2207 Computer Organization and Architecture Laboratory

Teaching Scheme: **Examination Scheme:**

Practical : **2** Hrs/Week Practical: **25** Marks

Credits: **1**

Course Objectives:

1. To understand configuration of Computer Systems
2. To understand fundamental working of Computer Systems.
3. To study architecture and features of 8086 microprocessor
4. To learn interfacing of 8086 microprocessor with I/O peripherals

Course Outcomes:

On completion of the course, student will be able to –

1. Write Assembly Language Programs to perform numeric operations.
2. Write Assembly Language Programs to perform string operations.
3. Interface various I/O peripherals with microprocessor.
4. Understand the internal architecture of modern processors.

Suggested List of Laboratory Assignments

1. Write Assembly Language Program (ALP) for
   a) Addition and subtraction of 8 bit numbers. OR
   b) Program to count negative numbers from signed numbers either stored in memory or given by user. OR
   c) Ascending/descending sort
2. Write ALP to convert 4-digit Hex number into its equivalent BCD number and 4-digit BCD number into its equivalent HEX number.
3. Write ALP to perform following operation on string:
   a) Find and display length
   b) Display reverse
   c) Check whether string is palindrome or not.
   d) Concatenation of two strings
   e) Find number of words
   Display proper strings to prompt the user while accepting the input and displaying the result.
4. Write ALP to interface 8255 (PPI) with 8086
5. Write ALP to interface 8251 (Serial Interface) with 8086
6. Write ALP to interface 8254/8253(Timer/Counter) with 8086
7. Write ALP to interface 8259 (Programmable interrupt Controller) with 8086
8. Study Assignment: Explain architecture of Quad core Processor in detail with an application

Text Books:

1. Douglas Hall, “Microprocessors and Interfacing, Programming and Hardware”, McGraw-Hill, ISBN: 0–07–100462–9

Reference Books:

1. Intel Manual

2. W. Stallings, “Computer Organization and Architecture: Designing for Performance”, 8th Edition, Prentice Hall of India, 2010, ISBN 13: 978-0-13-607373-4

IT 2208 Object Oriented Programming Laboratory

Teaching Scheme: **Examination Scheme:**

Practical: **2** Hrs/Week End-Semester: **50** Marks

Credits: **1**

Course Objectives:

The students should be able to understand abstraction

The students should be able to understand the encapsulation

The students should be able to understand the inheritance and polymorphism.

The students will be able to convert a small problem description in an object oriented code

Course Outcomes:

1. The students will be able to abstract required properties and behavior of a class from a description and implement them in java.
2. The students will be able to apply inheritance to a given problem description and implement them in java.
3. The students will be able to derive encapsulation and polymorphic behavior from a given problem description and implement them in java.
4. The students will be able to determine all applicable object oriented features from the given description and implement them in java.

List of assignments

1. Convert the given description into an object oriented language code. An employee has an employeeID, name. Display the data for five employees

2. Convert the given description into an object oriented language code. An employee has an employeeID, name. Every employee has a basic pay and a joining date. Display the data for five employees

3. Convert the given description into an object oriented language code. An employee has an employeeID, name and salutation. Every employee has a basic pay and a joining date. Display the data for five employees

4. Convert the given description into an object oriented language code. An employee has an employeeID, name, salutation and an address. The address has apartment number, apartment name, road, city, state and pincode. Every employee has a basic pay and a joining date. Display the data for five employees

5. Convert the given description into an object oriented language code. An employee has an employeeID, name, salutation and an address. The address has apartment number, apartment name, road, city, state and pincode. Each employee gets a monthly salary. A team leader is an employee and a software engineer is an employee. The team leader gets the monthly salary as basic pay, DA as 70 percent of basic pay and traveling allowance as 15 percent of basic pay. The software engineer gets a monthly salary as basic pay, Da as 35 percent of basic pay and traveling allowance as 12 percent of basic pay.

6. Convert the given description into an object oriented language code. An employee has an employeeID, name, salutation and an address. The address has an apartment number, apartment name, road, city, state and pincode. Each employee gets a monthly salary. A team leader is an employee and a software engineer is an employee. The team leader gets the monthly salary as basic pay, DA as 70 percent of basic pay and traveling allowance as 15 percent of basic pay. The software engineer gets a monthly salary as basic pay, Da as 35 percent of basic pay and traveling allowance as 12 percent of basic pay. Now, it is policy of the company that every software engineer will get an add on compensation if she works for more than 8 hours in a day. The compensation is calculated as Rs 200.00 per hour. If a team lead works for more than 8 hours in a day, she gets an add on compensation as Rs 600.00 for a slab of 4 hours. Incorporate this in the code.

7. Convert the following description in an object oriented language using abstraction, has a relationship, is a relationship, encapsulation and data hiding
A bank issues many credit cards. Each credit card has a credit card no. It has a list of purchases associated with it. Every purchase made using the credit card has date of purchase, amount of purchase and pay back points for that purchase. The credit card has the total payback points accumulated across all the purchases made. The policy for adding the payback points for every purchase is as follows

Sr No	**Date of purchase**	**Quarter**	**Pay back points**
1	1st Jan to 31st Mar	First	1 payback point for every 200 Rs purchase
2	1st April to 30th June	Second	1 payback point for every 150 Rs purchase
3	1st July to 30th Sept	Third	1 payback point for every 100 Rs purchase
4	1st Oct to 31st Dec	Fourth	1 payback point for every 80 Rs purchase

Calculate the total payback points for the following details
Credit Card = 123456789000

Date of purchase	Purchase amount
29rd March	20000.00
10th July	30000.00
15th Oct	15000.00
24th Dec	10000.00

Convert the following description in an object oriented language using abstraction, has a relationship, is a relationship, encapsulation and data hiding
An account has an accountNo, balance and an account holder. An account holder has a name and an address. Address has apartment number, apartment name, road, city, state and pincode. An amount can be withdrawn from an account, deposited to an account or transferred from one account to other account. A saving account is an account. A current account is an account. A saving account gets an interest from the bank with an annual interest rate of 3.5 percent. This interest gets added to the balance amount. A current account is charged with a commission by the bank. The commission is charged annually with a rate of 2.5 percent. This commission gets deducted from the balance of the current account. Create one saving accounts with two deposits and one withdrawal. Create second saving accounts with one deposit and two withdrawals. Create third saving accounts with one deposit, one withdrawal and a transfer to first account. Create fourth account as current account with one deposit, three withdrawals and commission for two years.

Text Books:

1. Kathy Sierra, **'OCA / OCP Java SE 7 Programmer I & II Study Guide (Exams 1Z0-803 & 1Z0-804), Chapter 1 and 2** *Oracle press (2017)*

Reference Books:

1. Khalid A Mughal, **'Programmer's Guide to Java Certification: A Comprehensive Primer',** *Oracle press (2017)*

Autonomous Programme Structure of
Third Year B Tech Information Technology
Academic Year 2019-2020

Third Year B. Tech. Information Technology Semester – I										
Course Code	Course Title	Teaching Scheme Hours /Week			Examination Scheme				Marks	Credit
		Lecture	Tutorial	Practical	In Semester	End Semester	Oral	Practical		
IT 3101	Database Management Systems	3	0	0	50	50	0	0	100	3
IT 3102	Theory of Computations	3	1	0	50	50	0	0	100	4
IT 3103	Machine Learning	3	1	0	50	50	0	0	100	4
OEHS 3101	Open Elective-I	3	0	0	50	50	0	0	100	3
PEIT 3101	Program Elective- I	3	0	0	50	50	0	0	100	3
IT 3104	Database Management Systems Laboratory	0	0	4	25	0	0	25	50	2
IT 3105	Machine Learning Laboratory	0	0	4	25	0	0	25	50	2
PEIT 3106	Program Elective I Laboratory	0	0	2	0	0	25	0	25	1
AC 3101	Audit Course	0	0	2	0	0	0	0	0	0
	Total	**15**	**2**	**12**	**300**	**250**	**25**	**50**	**625**	**22**
	Grand Total	**29**			**625**				**625**	**22**

OEHS 3101 Elective 1

1. Entrepreneurship Development
2. Introduction to Digital Marketing
3. Intellectual Property Rights
4. Project Management

PEIT 3101 Program Elective-I
PEIT 3106 Program Elective-I Laboratory

1. Computer Graphics and Animation
2. Artificial Intelligence
3. Business Intelligence
4. Object Oriented Modeling and Design

AC 3101: Audit Course: Employability Skills Development

MKSSS's Cummins College of Engineering for Women Karvenagar, Pune-411052

DEAN ACADEMICS
MKSSS's Cummins College of Engineering for Women Karvenagar, Pune-411052

Principal
MKSSS's Cummins College of Engg For Women, Karvenagar, Pune-52

APPROVED BY
Governing Body Members
MKSSS's Cummins College of Engineering for Women Karvenagar, Pune-411052

IT 3101 Database Management Systems

Teaching Scheme:
Lectures: **3** hrs/week
Tutorial: NIL

Examination Scheme:
In-Semester: **50** marks
End-Semester: **50** marks
Credits: **3**

Prerequisites: Data structures

Course Objectives:

Familiarize students with

1. Concepts and applications of database management.
2. Different models and normalization used for database design.
3. Query languages in databases.
4. Basic issues of database design and utilization.

Course Outcomes:

Students should be able to

1. Identify basic purpose and functions of database management system.
2. Build appropriate database schema for the given application.
3. Make use of query commands and concurrency control protocols.
4. Analyze database for given problem domain.

Unit – I: Introduction to DBMS **(07)**

Database Concepts, Database System Architecture, Data Models, entity, attributes, relationships, constraints, keys, E-R Model, conventions, EER Model, converting ER/EER diagram into tables. Relational Model, Attributes and Domains, Referential Integrities. Relational Algebra: Basic Operations

Unit – II: Database Design and SQL **(07)**

Database Design, Functional Dependency, Purpose of Normalization, Data Redundancy, Anomalies. Normal forms 1NF, 2NF, 3NF, BCNF, 4NF and 5NF. Introduction to SQL, SQL Data Types, DDL, DML and DCL queries, Views, Indexes, Null handling, Nested Queries. PLSQL. Query optimization

Unit – III: Database Transactions **(07)**

Basic concept of a Transaction, Transaction Management, Properties of Transactions, Concept of Schedule, Serial Schedule, Serializability, Conflict and View, Cascaded Aborts, Recoverable and Non recoverable Schedules.

Unit – IV: Concurrency control and Advanced Database Architectures **(07)**

Concurrency Control, Locking Methods, Deadlocks, Protocols, Recovery Methods, Database Architectures, Centralized and Client-Server Architectures, 2 Tier and 3 Tier Architecture, Indexing and hashing, Parallel Databases, Distributed Databases.

Unit – V: No SQL and semi structured Data Management **(07)**

Introduction to Big Data, No SQL Databases, MongoDB, Map reduce. XML Databases, DTD, XML Schemas, XQuery, XPath. JSON

Unit – VI: Data Warehousing and Data Mining **(07)**

Data Warehousing, Architecture and features of Data Warehouse, ETL Process, OLAP. Data Mining, Knowledge Discovery, Data Mining techniques, Applications of data mining.

Text Books:

1. Silberschatz A., Korth H., Sudarshan S, Database System Concepts, McGraw Hill Publication, ISBN- 0-07-120413-X, Sixth Edition.
2. Ramez Elmasri,Shamkant B. Navathe**,** Fundamentals of Database Systems, Pearson Publication, ISBN-13: 978-0-136-08620-8

Reference Books:

1. S. K. Singh, Database Systems: Concepts, Design and Application, Pearson Publication, ISBN-978-81- 317-6092-5.
2. C J Date, An introduction to Database Systems, Addition-Wesley.
3. Data base Management Systems, Raghurama Krishnan, Johannes Gehrke, TATA McGrawHill, 3rd Edition, 2003.
4. Reema Thareja, Data warehousing, Oxford University Press. ISBN 0195699610.

IT 3102 Theory of Computations

Teaching Scheme:
Lectures: **3** hrs/week
Tutorial: **1** hr/week

Examination Scheme:
In-Semester: **50** marks
End-Semester**: 50** marks
Credits: **4**

Prerequisites: Discrete structures

Course Objectives:

Familiarize students with

1. Abstract computing models.
2. Types and applications of formal grammars
3. Application of Theory of Computer Science in System Programming

Course Outcomes:

Students should be able to

1. Construct abstract computing models
2. Apply the concepts of formal grammars
3. Analyze Decidable Languages and Reducibility
4. Evaluate computing models.

Unit – I Fundamentals (07)

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

Unit – II Finite Automata with application (07)

NFA with e transitions - Significance, acceptance of languages, Equivalence between NFA with and without e transitions, minimization of FSM, equivalence between two FSM's, Finite Automata with output- Moore and Mealy machines.

Lexical analyzer as an application of Finite Automaton. Introduction to Lex tool

Unit – III Regular Expression and Grammar Formalism (07)

Regular expressions: Identity rules, Constructing finite Automata for a given regular expressions, Conversion of Finite Automata to Regular expressions. Pumping lemma of regular sets, closure properties of regular sets

Introduction to Grammar: derivation trees, sentential forms. Right most and leftmost derivation of strings, Chomsky hierarchy

Unit – IV Regular Grammar with application (07)

Regular grammars-right linear and left linear grammars, equivalence between regular grammar and FA, inter conversion, Parsing techniques, Top-down parsing, Bottom-up parsing

Recursive descent parser as an application of Regular Grammar. . Introduction to YACC tool

Unit – V Context free grammars and Push down automata (07)

Context Free Grammars-Ambiguity in context free grammars. Minimization of Context Free Grammars. Normal Forms Chomsky Normal Form, Greibach Normal Form, conversion to CNF and GNF

Push down automata- definition, model, acceptance of CFL, Acceptance by final state and acceptance by empty state and its equivalence. Equivalence of CFL and PDA, inter conversion,

Unit – VI Turing Machine (07)

Turing Machine, definition, model, design of TM, Computable functions, recursively enumerable languages. Church's hypothesis, counter machine, types of Turing machines, Universal Turing Machine, decidability/un decidability of problems, Halting problem Correspondence problem, Turing reducibility

Modularized programming concept as an application of Turing machines

Text Books

1. Daniel I.A. Cohen, "Introduction to Computer Theory" Wiley-India, ISBN: 978-81-265-1334-5
2. Vivek Kulkarni, "Theory of Computation", Oxford University Press, ISBN-13: 978-0-19-808458-7.
3. D.M. Dhamdhere, "Systems Programming and Operating Systems", Tata McGraw-Hill, ISBN-13:978-0-07-463579-7

Reference Books

1. John C. Martin, "Introduction to language and theory of computation", Tata McGraw Hill, Third edition, ISBN 0-07-049939-X
2. Hopcroft Ulman, "Introduction To Automata Theory, Languages And Computations", Pearson Education Asia, 2nd Edition
3. E V Krishnamurthy, "Introduction to Theory of Computer Science", EWP Second 2nd Edition.
4. John J Donovan , "Systems Programming", Tata McGraw-Hill Edition 1991, ISBN 0-07-460482-1

IT 3103 Machine Learning

Teaching Scheme:
Lectures: **3** hrs/week
Tutorial: **1** hr/week

Examination Scheme:
In-Semester: **50** marks
End-Semester: **50** marks
Credits: **4**

Prerequisites: Linear Algebra and Calculus, Probability Basics

Course Objectives:

Familiarize students with

1. Basic learning algorithms and techniques
2. Applications of machine learning
3. Usage of large data sets

Course Outcomes:

Students should be able to

1. Identify different tasks in machine learning
2. Explain wide variety of learning algorithms and techniques
3. Apply proper learning algorithm to data depending on the task
4. Perform evaluation of learning algorithms

Unit – I Introduction to Machine Learning (07)

Introduction: What is Machine Learning, Examples of Machine Learning applications, Training versus Testing, Positive and Negative Class, Cross validation

Types of Learning: Supervised, Unsupervised and Semi-Supervised Learning

Features: Types of features-Continuous, Discrete, Nominal, Ordinal. Extraction of Features from text document and image

Unit – II Classification (07)

Binary and Multiclass Classification: Assessing Classification Performance, Performance of multi-class classification

Linear and Non-linear Models: Perceptron, Support Vector Machines (SVM), Soft Margin SVM, Kernel methods for non-linearity

Unit – III Regression and Generalization (07)

Regression: Assessing performance of Regression – Error measures, Overfitting and Underfitting, Catalysts for Overfitting

Linear Models: Least Square method,Univariate Regression

Theory of Generalization: Bias and Variance Dilemma, Training and Testing Curves

Case Study of Polynomial Curve Fitting

Unit – IV Distance Based Models (07)

Neighbors and Examples, Distance Measures: Euclidian, Manhattan, Minkowski, Hamming. Nearest Neighbor Classification (kNN), Distance based clustering algorithms - K-means, K-medoid, DBScan, Hierarchical Clustering: Single, Complete, Average and Centroid Linkage

Unit – V Rule and Tree based Models (07)

Rule Based Models: Frequent Itemsets, Association rules mining – Apriori Algorithm, Confidence and Support parameters

Tree Based Models: Impurity Measures – Entropy, Gini Index, Information Gain. Decision Trees, ID3

Unit – VI Probabilistic Models (07)

A brief overview of Probability Theory – Discrete Random Variable, Joint Probability, Conditional Probability, Joint Probability, Bayes' Theorem, Independence Assumption, Naïve Bayes Classification, Probabilistic Models with Hidden Variables: Expectation Maximization

Text Books

1. Ethem Alpaydin: Introduction to Machine Learning, PHI 2nd Edition-2013.
2. Peter Flach: Machine Learning: The Art and Science of Algorithms that Make Sense of Data, Cambridge University Press, Edition 2012.

Reference Books

1. C. M. Bishop: Pattern Recognition and Machine Learning, Springer 1st Edition-2013.
2. Ian H Witten, Eibe Frank, Mark A Hall: Data Mining, Practical Machine Learning Tools and Techniques, Elsevier, 3rd Edition.
3. Jiawei Han, Micheline Kamber: Data Mining: Concepts and Techniques, Morgan Kaufmann Publishers, 3rd Edition, July 2011.
4. Kevin Murphy: Machine Learning – A Probabilistic Perspective, MIT Press, 2012.
5. Parag Kulkarni, Prachi Joshi: Artificial Intelligence: Building Intelligent Systems, Prentice Hall of India.

PEIT 3101 Computer Graphics and Animation

Teaching Scheme:
Lectures: **3** hrs/week

Examination Scheme:
In-Semester: **50** marks
End-Semester: **50** marks
Credits: **3**

Prerequisites: Basics of Programming, Essential mathematics in geometry and trigonometry, Vectors and Matrices

Course Objectives:
Familiarize students with
1. Basic concepts of computer graphics
2. Basic primitives and objects in computer graphics
3. Various methods and techniques used in computer graphics
4. Applications of computer graphics in animation and gaming

Course Outcomes:
Students should be able to
1. Identify geometrical formulas and algorithms to draw computer graphics primitives
2. Use mathematics to transform computer graphics objects
3. Apply various techniques to achieve desired image manipulation.
4. Design algorithmic logic to solve complex problem like gaming

Unit – I Basic Concepts (07)
Introduction: Basics of graphics systems, Raster scan & Random scan displays, basic display processor. **Display Files**: display file structure, algorithms and display file interpreter. Primitive operations.
Plotting Primitives: Scan conversions, line segments, vectors, pixels and frame buffers, vector generation. **Introduction to OpenGL:** Basic OpenGL syntax, display-window management using GLUT, functions.

Unit – II Drawing and Filling Graphics Primitives (07)
Line and Circle drawing Algorithms: DDA, Bresenham's, Midpoint.
Character Generation: Stroke Principle, Starburst Principle, Bit map method, aliasing and anti-aliasing
Polygon: Polygon and its types, inside test, polygon filling methods: Seed fill, Scan Line, Flood fill and Boundary fill

Unit – III Geometric Transformations (07)
2D Geometric Transformations: Translation, scaling, rotation, reflection, shearing, matrix representation and homogeneous coordinate system, Composite transformations
3D Geometric Transformations: Translation, scaling, rotation, rotation about X, Y, Z and arbitrary axis reflection about XY, YZ, XZ and arbitrary plane.

Unit – IV Segments, Windowing and Clipping (07)
Segment: Introduction, Segment table, Segment creation, closing, deleting and renaming, Visibility
Windowing: Concept of window and viewport, viewing transformations
Line Clipping: Cohen Sutherland Method, Midpoint subdivision method

Polygon Clipping: Sutherland Hodgman method for clipping convex and concave polygon

Unit – V Shading and Animation (07)

Shading: Halftoning, Gouraud and Phong Shading

Computer Animation: Design of Animation sequences, General Computer Animation functions, Computer Animation Languages, Key-frame Systems, Motion Specifications.

Unit – VI Gaming (07)

Gaming platforms: Graphics Memory Pipeline, Block diagram of NVIDIA workstation and i860 Introduction to OpenGL ES

Interactive Graphics & usage of the tools of computer graphics: 3D Studio and Maya

2D games : Snake game

Text Books

1. D. Hearn, M. Baker, "Computer Graphics with OpenGL", 4th Edition, Pearson Education, 2014,ISBN 978-93-325-1871-1
2. S. Harrington, "Computer Graphics", 2nd Edition, McGraw-Hill Publications, 1987, ISBN 0 – 07 – 100472 – 6.

Reference Books

1. D. Rogers, "Procedural Elements for Computer Graphics", 2nd Edition, Tata McGraw-Hill Publication, 2001, ISBN 0 – 07 – 047371 – 4.
2. J. Foley, V. Dam, S. Feiner, J. Hughes, "Computer Graphics Principles and Practice", 2nd Edition,Pearson Education, 2003, ISBN 81 – 7808 – 038 – 9.
3. D. Rogers, J. Adams, "Mathematical Elements for Computer Graphics", 2nd Edition, Tata McGraw-Hill Publication, 2002, ISBN 0 – 07 – 048677 – 8.
4. Zhigang Xiang, Roy Plastock, "Computer Graphics", Schaum's Series outlines
5. F.S. Hill JR, "Computer Graphics Using Open GL", Pearson Education
6. Samuel R. Buss, "3D Computer Graphics", Cambridge University Press

PEIT 3101 Artificial Intelligence

Teaching Scheme:
Lectures: **3** hrs/week

Examination Scheme:
In-Semester: **50** marks
End-Semester: **50** marks
Credits: 3

Prerequisites: Discrete mathematics, basic probability theory and statistics
Knowledge of any programming language and data structures

Course Objectives:
Familiarize students with

1. The basic principles and applications of Artificial Intelligence.
2. Concepts of problem solving and knowledge representation
3. Concepts of planning and learning

Course Outcomes:
Students will be able to:

1. Assess underlying AI concepts and their usage.
2. Implement classical Artificial Intelligence techniques, such as search algorithms, minimax algorithm, and neural networks.
3. Represent knowledge using logic and infer new facts from it.
4. Apply Artificial Intelligence techniques for problem solving.

Unit – I Artificial Intelligence (07)
Introduction -What is AI? The Foundations of Artificial Intelligence,
Intelligent Agents: Agents and Environments, Good Behavior: The Concept of Rationality, The Nature of Environments, The Structure of Agents.

Unit – II Problem Solving (07)
Problem Solving: Solving Problems by Searching, heuristic search techniques, constraint satisfaction problems: Constraint Propagation, Backtracking Search for CSPs,
Game Playing: Minimax algorithm, alpha-beta pruning.

Unit – III Knowledge Representation (07)
Logical Agents: Knowledge-Based Agents, Propositional logic, First-order Logic,
Knowledge Representation: Ontological Engineering, Categories and Objects, Events, Reasoning Systems for Categories, Reasoning with Default Information

Unit – IV Planning (07)
Planning: Definition of Classical Planning, Algorithm for Planning as State-Space Search, Planning Graphs, Other Classical Planning Approaches, Analysis of Planning Approaches
Planning and Acting in the Real World: Time, Schedules, and Resources.

Unit – V Reasoning and Learning (07)
Quantifying Uncertainty: Acting under Uncertainty, Basic Probability Notation, Reasoning: Probabilistic Reasoning, Making Simple Decisions: Decision Networks
Learning: Forms of Learning, Supervised Learning, Learning Decision Trees, Ensemble Learning, Knowledge in Learning, Learning Probabilistic Models, Reinforcement Learning.

Unit – VI Artificial Neural Network (07)

Units in neural networks, Neural Network structures, Single layer feed-forward neural networks, Multilayer feed-forward neural networks, Learning in multilayer networks, Learning neural network structures.

Text Books

1. Stuart Russell and Peter Norvig, Artificial Intelligence: A Modern Approach, 3rd Edition, Prentice Hall.

Reference Books

1. E. Rich and K.Knight, Artificial Intelligence, Tata McGraw Hill
2. Charniack and D. Mcdermott, Artificial Intelligence, Addison Wesley
3. George F. Luger , "Artificial Intelligence: Structures and Strategies for Complex Problem Solving", Pearson
4. N.P. Padhy, "Artificial Intelligence And Intelligent Systems", Oxford University Publishers
5. Parag Kulkarni, Prachi Joshi: Artificial Intelligence: Building Intelligent Systems, Prentice Hall of India.
6. Toby Segaran, Programming Collective Intelligence, O'Reilly

PEIT 3101 Business Intelligence

Teaching Scheme:
Lectures: **3** hrs/week

Examination Scheme:
In-Semester: **50** marks
End-Semester: **50** marks
Credits: 3

Prerequisites: Linear algebra, probability basics

Course Objectives:

Familiarize students with

1. The role of Business Intelligence in various business applications
2. Methods of data processing and modeling
3. Importance of visualization and reporting in business
4. Decision making process using Business Intelligence

Course Outcomes:

Students will be able to:

1. Identify business problems to provide BI solutions
2. Use data transformation and modeling concepts for building data warehouse
3. Analyze and visualize dimensional models for reporting.
4. Comprehend different BI trends and their applications.

Unit – I Introduction (07)

Concepts of Data, Information, and Knowledge, Design and implementation aspect of OLTP and OLAP/Data Warehouse, Business Intelligence(BI) Concepts and definitions, BI architectural models (Top-down and bottom-Up), Business Applications of BI, Role of Data warehouse in BI, BI system components

Unit – II Dimensional Modeling And Data Warehouse Design (07)

Star schema, Snow flake schema, and Fact Constellation schema, Grain of dimensional model, transactions, Recurring Snapshots, Accumulating Snapshots, Dimensions (SCD types, conformed dimensions), Facts (additive, semi-additive, non-additive), Junk dimensions, conformed dimensions, Bridge tables

Unit – III ETL (07)

Data Quality, Data profiling, Data enrichment, data duplication, Data cleaning, ETL Architecture and what is ETL, Extraction concept and Change data capture, Transformation concept, Loading concept, Initial and Incremental loading, Full loading, late arriving facts, data staging, Data marts

Unit – IV Reporting (07)

Metadata Layer, Presentation Layer, Data Layer, Use of different layers and overall Reporting architecture, Various report elements such as Charts, Tables, Materialized views, Query rewrite, Ad-hoc reports, Security: report level, data level (row, column),Scheduling.

Unit – V Analytics And Data Visualization (07)

Analytics: Application of Analytics concepts in Business Intelligence, Clustering with K-Means, Classification with Decision tree, In-Memory Analytics and In-DB Analytics

Data visualization: Types of data visualization, Techniques for visual data representations, data

Visualization tools- Tableau, Dashboards, **Case study:** Credit card fraud detection, click stream analysis

Unit – VI Recent Trends (07)

Introduction to Big Data, DW appliances, Types of BI: Real time BI, Operational BI, Embedded BI, Agile BI, Smart change data capture using log based techniques

Case Study: BI for sales force management, Social BI systems

Text Books

3. Ralph Kimball, Joe Caserta, "The Data warehouse ETL Toolkit", Publisher: Wiley
4. Jiawei Han, Micheline Kamber, Jian Pei "Data Mining: concepts and techniques", 2nd Edition, Publisher: Elsevier/Morgan Kaufmann.

Reference Books

1. Ralph Kimball, Margy Ross, "The Data Warehouse Toolkit", 3rd edition, Publisher: Wiley
2. Reema Thareja, "Data Warehouse", Publisher: Oxford University Press.
3. William Inmon, "Building the Data Warehouse", Wiley publication 4th edition.

PEIT 3101 Object Oriented Modeling and Design

Teaching Scheme:
Lectures: **3** hrs/week

Examination Scheme:
In-Semester: **50** marks
End-Semester: **50** marks
Credits: 3

Prerequisites: Object Oriented Programming

Course Objectives:
Familiarize students with

1. Need of Object oriented Modeling and design
2. Unified Modeling language diagrams for representing the modeling
3. Purpose of every UML diagram for showing an aspect of modeling
4. Applying the analysis and design methodology to a moderate complex system

Course Outcomes:
Students should be able to

1. Identify the classes, attributes for moderately complex systems.
2. Co-relate the classes with has-a or is-a relationship.
3. Refine the relationships to remove the redundancy.
4. Represent the design using UML diagrams.

Unit – I Software Complexity Understanding the challenges OOAD can address (07)
Software Complexity, Object Model, Classes and Objects, Identification approaches using OOAD

Unit – II Use Case model and Activity Model (07)
Identifying use cases from requirements, Representing business flow as an activity diagram.
Relationships in use cases as extends, generalization, includes.
Notations of Activity diagrams an activity, join, fork, decision node, merge node, swim lane

Unit – III Domain Analysis Model (07)
Relationships among classes as association, aggregation, composite, generalization, dependency, realization, association class , binary association, ternary association, multiple inheritance

Unit – IV Interaction Model (07)
Realization of use cases, use case specifications, representing use case and a small class diagram, showing the interaction among classes as sequence diagram

Unit – V SMART design Principles Architecture design (07)
Overview of SMART design principles, design pattern and software architecture design

Unit – VI Organization and deployment of Software (07)
Purpose of package diagram, component diagram and deployment diagram

Text Books

1. Grady Booch, “Object Oriented Analysis and Design using applications”, 3rd Edition,
2. Russ Miles, “Learning UML 2.0 “, O’Reilly

Reference Books

1. Grady Booch, “UML 2.0 use Guide” OMG Group

IT 3104 Database Management Systems Laboratory

Teaching Scheme:
Practical: **4** hrs/week

Examination Scheme:
In-Semester: **25** marks
Practical: **25** marks
Credits: 2

Prerequisites: Data structures

Course Objectives:

Familiarize students with

1. Implementation of fundamental concepts of database management
2. Use of database management.
3. SQL database system and PLSQL.
4. NOSQL database system.

Course Outcomes:

Students should be able to

1. Make use of database systems.
2. Design and use database schema for given application.
3. Develop a database application with suitable front end.
4. Implement SQL and NOSQL commands.

Group A: Introduction to Databases (Study assignment- Any one)

1. Study and design a database with suitable example using following database systems:
    a. Relational: SQL / PostgreSQL / MySQL
    b. Columnar: Hbase
    c. Document: MongoDB / CouchDB
    d. Graph: Neo4J

    Compare the different database systems based on points like efficiency, scalability, characteristics and performance.
2. Study the SQLite database and its uses. Also elaborate on building and installing of SQLite

Group B: SQL and PL/SQL (Minimum 6)

1. Design any database with at least 3 entities and relationships between them. Apply DCL and DDL commands. Draw suitable ER/EER diagram for the system.
2. Execute DDL statements which demonstrate the use of views. Try to update the base table using its corresponding view. Also consider restrictions on updatable views and perform view creation from multiple tables.
3. Design and implement a database and apply at least 10 different DML queries for the following task. For a given input string display only those records which match the given pattern or a phrase in the search string.
4. Execute the aggregate functions like count, sum, avg etc. on the suitable database. Use group by and having clauses. Retrieve the data from the database based on time and date functions.

5. Implement nested sub queries. Perform a test for set membership (in, not in), set comparison (=some, >=some, <all etc.) and set cardinality (unique, not unique).
6. Write and execute suitable database triggers.
7. Write and execute PL/SQL stored procedure and cursor to perform a suitable task on the database.

Group C: NoSQL and Semi structured Databases (Minimum 3)

1. Create a database with suitable example using MongoDB and implement
    - Inserting and saving document
    - Removing document
    - Updating document
2. Execute at least 15 different queries on any suitable MongoDB database that demonstrates following querying techniques:
    - find and findOne
    - Query criteria
    - Type-specific queries
    - $ where queries
    - Create and drop different types of indexes
3. Implement Map reduce example using Mongo DB.
4. Design and implement XML/JSON database.

Group D: Mini Project / Database Application Development

Student group preferably of size 4 students should decide the statement and scope of the project which will be refined and validated by the faculty.

Choose database as per the requirement of the mini project. Draw and normalize the design up to at ER Diagram with normalization in case of back end as RDBMS. Design front end using any open source technology and perform connectivity to the database. Implement suitable database operations along with business logic, validations, reports etc.

IT 3105 Machine Learning Laboratory

Teaching Scheme:
Practical: **4** hrs/week

Examination Scheme:
In-Semester: **25** Marks
Practical: **25** marks
Credits: **2**

Prerequisites: Linear Algebra and Calculus, Probability Basics

Course Objectives:

Familiarize students with

1. Various tasks in Machine Learning
2. Different Machine Learning algorithms
3. Applications of machine learning algorithms for accomplishing given tasks

Course Outcomes:

Students should be able to

1. Apply proper learning algorithm to data depending on the task
2. Compare different learning algorithms performing similar tasks
3. Use large data sets
4. Evaluate the models

Implementation of programs to be done in Python

1. Classify data using Linear Support Vector Machine algorithm
2. Predict values using Linear Regression with one independent variable and one dependent variable
3. Cluster data using k-means algorithm for clustering
4. Identify frequent item item-sets using Apriori algorithm
5. Classify data using Naïve Bayes Classification algorithm
6. Build a small application using machine learning concepts.

Text Books

1. Andreas Muller and Sarah Guido: Introduction to Machine Learning with Python, O'Reilly, 2017
2. Michael Bowles: Machine Learning in Python, Wiley, 2018

Reference Books

1. Ian H Witten, Eibe Frank, Mark A Hall: Data Mining, Practical Machine Learning Tools and Techniques, Elsevier, 3rd Edition.
2. Jiawei Han, Micheline Kamber: Data Mining: Concepts and Techniques, Morgan Kaufmann Publishers, 3rd Edition, July 2011.
3. C. M. Bishop: Pattern Recognition and Machine Learning, Springer 1st Edition-2013.
4. Ethem Alpaydin: Introduction to Machine Learning, PHI 2nd Edition-2013.
5. Peter Flach: Machine Learning: The Art and Science of Algorithms that Make Sense of Data, Cambridge University Press, Edition 2012.

PEIT 3106 Computer Graphics Laboratory

Teaching Scheme:
Practical: **2** hrs/week

Examination Scheme:
Oral: **25** marks

Credits: 1

Prerequisites: Basics of Programming, Data Structures, Algorithms, Geometry, Trigonometry, Vectors and Matrices

Course Objectives:
Familiarize students with

1. Various methods and techniques used in computer graphics
2. Applications of computer graphics in animation and gaming

Course Outcomes:
Students should be able to

1. Apply mathematics and algorithms to draw computer graphics primitives.
2. Apply graphics data manipulation in an application.
3. Implement programs using different computer graphics algorithm.
4. Make use of OPEN GL to implement programs

List of Assignments

1 Get Familiar with basic OpenGL environment, display-window management using GLUT, OpenGL functions.

2 Write a function in OpenGL on Linux Platform to draw a Line using DDA/ Bresenham's Line Drawing Algorithm. Call the Function to draw any pattern consisting at least 10 function calls.

3 Write a function in OpenGL on Linux Platform to draw a circle using Midpoint Circle Drawing Algorithm. Call this function at least 6 times to draw any pattern. User should only give centre coordinates and radius. Rest should be drawn automatically

4 Write a program in OpenGL on Linux Platform to draw chess board using any Line drawing algorithm and fill alternate blocks using flood fill algorithm

5 Write a program in OpenGL on Linux Platform to draw a flag using any Line drawing algorithm and fill it using scanline polygon filling algorithm.

6 Write a program in OpenGL on Linux Platform to for drawing a polygon and perform following 2DTransformations on Triangle.
Translation,Scaling, Rotation

7 Write a program in OpenGL on Linux Platform to clip a Line using Cohen Sutherland Outcode Method.

8 Write a program in OpenGL on Linux Platform to clip a Polygon using SutherlandHodgman Polygon Clipping.

9 Write a program in OpenGL on Linux Platform to animate a scene for "Moving Car".

10 Write a program to design a game using computer graphics basic techniques and OpenGL

PEIT 3106 Artificial Intelligence Laboratory

Teaching Scheme:
Practical: **2** hrs/week

Examination Scheme:
Oral: **25** marks
Credits: **1**

Course Objectives:
Familiarize students with
1. Basic implementation of AI concepts.
2. Current Trends in AI.

Course Outcomes:
Students will be able to
1. Implement AI core concepts using AI algorithms.
2. Identify appropriate AI techniques for development of applications.
3. Apply basic principles of AI towards problem solving, knowledge representation and learning.
4. Gain basic understanding of various AI applications in intelligent and expert systems, artificial neural networks and other machine learning techniques.

List of Assignments (Any 5)

1. Implement A* algorithm for any of the following problems: a) 8 puzzle b) Missionaries and Cannibals c) Blocks World Problem
2. Solve 8-queens problem using backtracking.
3. Implement a program to solve constraint satisfaction problem using any searching technique.
4. Implement minimax algorithm using alpha-beta pruning.
5. Implement the code for decision tree learning.
6. Implement Truth Maintenance System.
7. Implement Neural network to understand backpropagation.

Group Assignment
1. Develop application such as but not limited to
   a) Chatbot
   b) Interactive Sudoku solver
   c) Stock market predictor (offline past data)
   d) Face Recognition
   e) Captcha breakers
   f) Auto tagging of friends on social media
   g) Pac-Man

PEIT 3106 Business Intelligence Laboratory

Teaching Scheme:
Practical: **2** hrs/week

Examination Scheme:
Oral: **25** marks
Credits: **1**

Prerequisites: Linear Algebra and Calculus, Probability Basics, database concepts

Course Objectives:

Familiarize students with

1. BI tools and technologies
2. Data transformation techniques and modeling
3. Implementation aspects of business analytics and reporting

Course Outcomes:

Students should be able to

1. Identify the business problem and design BI solution
2. Analyze the model
3. Visualize large datasets
4. Implement BI application

Suggested list of laboratory assignments:

Given a Business Problem as Case Study design and build BI solution using BI concepts:

1. Perform dimension modeling and Execute ETL process for building data warehouse
2. Implement OLAP operations on given data set.
3. Visualize data using various charts in Tableau
4. Develop any one application

Text Books

1. Big Data, Black Book, DT Editorial services, 2015 edition
2. Jiawei Han, Micheline Kamber, Jian Pei "Data Mining: concepts and techniques", 2nd Edition, Publisher: Elsevier/Morgan Kaufmann

Reference Books

1. Ralph Kimball, Joe Caserta, "The Data warehouse ETL Toolkit", Publisher: Wiley

PEIT 3106 Object Oriented Modeling and Design Laboratory

Teaching Scheme:
Practical: **2** hrs/week

Examination Scheme:
Oral : **25** marks
Credits: **1**

Prerequisites: Object Oriented Programming

Course Objectives:

Familiarize students with

1. Modeling a complex system from various views
2. UML diagrams of conveying the model which in programming language independent

Course Outcomes:

Students should be able to

1. Identify the classes, attributes for a moderately complex systems
2. Co-relate the classes with has-a or is-a relationship
3. Refine the relationships to remove the redundancy
4. Represent the design using UML diagrams

List of Laboratory Assignments

Based on a system description given to a group of four students, they will be systematically analyzing and designing the system and drawing UML diagram to show the thought process via following assignments

1. Use Case Identification and Use case diagram
2. Activity diagram to show the business process
3. Class Diagram with advanced features
4. Sequence diagrams of each use case
5. State diagram
6. Package diagram and component diagram
7. Deployment diagram

Third Year B. Tech. Information Technology Semester – II										
Course Code	Course Title	Teaching Scheme Hours /Week			Examination Scheme				Marks	Credit
		Lecture	Tutorial	Practical	In Semester	End Semester	Oral	Practical		
IT 3201	Design and Analysis of Algorithms	3	1	0	50	50	0	0	100	4
IT 3202	Operating Systems	3	0	0	50	50	0	0	100	3
IT 3203	Software Engineering	3	1	0	50	50	0	0	100	4
PEIT 3201	Program Elective-II	3	0	0	50	50	0	0	100	3
PEIT 3202	Program Elective- III	3	0	0	50	50	0	0	100	3
IT 3204	Seminar	0	0	2	0	0	25	0	25	1
IT 3205	Programming Skills Development Laboratory	0	0	2	25	0	0	0	25	1
IT 3206	Operating Systems Laboratory	0	0	4	25	0	0	25	50	2
PEIT 3203	Program Elective III Laboratory	0	0	2	0	0	0	25	25	1
AC 3201	Audit Course	0	0	2	0	0	0	0	0	0
	Total	**15**	**2**	**12**	**300**	**250**	**25**	**50**	**625**	**22**
	Grand Total	**29**			**625**				**625**	**22**

Program Elective II
1. Advanced Computer Networks
2. Advanced Computer Architecture
3. Human Computer Interaction
4. Online course from Swayam

Program Elective III
1. Multimedia Computing
2. Natural Language Processing
3. Advanced Machine Learning
4. Systems Programming

AC 3201: Audit Course: Employability Skills Development

MKSSS's Cummins College of Engineering for Women Karvenagar, Pune-411052

DEAN ACADEMICS
MKSSS's Cummins College of Engineering for Women Karvenagar, Pune-411052

Principal
MKSSS's Cummins College of Engg For Women, Karvenagar, Pune-52

APPROVED BY
Governing Body Members
MKSSS's Cummins College of Engineering for Women Karvenagar, Pune-411052

IT 3201 Design and Analysis of Algorithms

Teaching Scheme:
Lectures: **3** hrs/week
Tutorial: **1** hour

Examination Scheme:
In-Semester: **50** marks
End-Semester: **50** marks
Credits: **4**

Prerequisites: Data structures

Course Objectives:

Familiarize students with

1. Algorithmic approaches for problem solving
2. Basics of computational complexity analysis
3. Various algorithm design strategies.
4. Different classes and solutions to problems such as P, NP etc.

Course Outcomes:

Students should be able to

1. Apply various algorithmic techniques to solve problem.
2. Determine computational complexity for various algorithms.
3. Apply appropriate algorithmic strategy for given problem.
4. Analyze and identify the class of the given problem and apply appropriate algorithms.

Unit – I: Introduction **(07)**

Analysis of Algorithm, Efficiency- Analysis framework, asymptotic notations. Proof Techniques, Introduction to Brute Force method & Exhaustive search, Analysis of Non-recursive and recursive algorithms: Solving Recurrences

Unit – II: Divide and conquer method and Greedy strategy **(07)**

Divide & Conquer method: Merge sort, Quick Sort. Binary search, Finding Max-Min, Large integer Multiplication, TOH. Greedy Method:MST for graph, Dijkstra's Algorithm, Fractional Knapsack problem, Job Sequencing.

Unit – III: Dynamic Programming **(07)**

General strategy, optimal substructure, 0/1 knapsack Problem, Chain matrix multiplication, Bellman-Ford Algorithm, Multistage Graph problem, Optimal Binary Search Trees, Travelling Salesman Problem.

Unit – IV: Backtracking **(07)**

General method, Recursive backtracking algorithm, Iterative backtracking method. 8-Queen problem, Sum of subsets, Graph coloring, Hamiltonian Cycle, 0/1 Knapsack Problem.

Unit – V: Branch and bound **(07)**

The method, Control abstractions for Least Cost Search, Bounding, FIFO branch and bound, LC branch and bound, 0/1 Knapsack problem – LC branch and bound and FIFO branch and bound solution, Traveling sales person problem

Unit – VI: Classes of algorithms **(07)**

Computational Complexity: Non Deterministic algorithms, The classes: P, NP, NP Complete, NP Hard, Satisfiability problem, NP Complete Problems, Parallel Algorithms, Randomized and approximation algorithms

Text Books:

1. Horowitz and Sahani, Fundamentals of computer Algorithms, Galgotia, ISBN 81-7371-612-9.
2. S. Sridhar, Design and Analysis of Algorithms, Oxford, ISBN 10 : 0-19-809369-1

Reference Books:

1. Thomas H Cormen and Charles E.L Leiserson, Introduction to Algorithm, PHI, ISBN: 81-203-2141-3.
2. R. C. T. Lee, SS Tseng, R C Chang, Y T Tsai, Introduction to Design and Analysis of Algorithms, A Strategic approach, Tata McGraw Hill, ISBN-13: 978-1-25-902582-2. ISBN-10: 1-25-902582-9.
3. Anany Levitin, Introduction to the Design & Analysis of Algorithm, Pearson, ISBN 81- 7758-835-4.
4. Gilles Brassard, Paul Bratle, Fundamentals of Algorithms, Pearson, ISBN 978-81-317-1244-3.

IT 3202 Operating Systems

Teaching Scheme:
Lectures: **3** hrs/week

Examination Scheme:
In-Semester: **50** marks
End-Semester: **50** marks
Credits: **3**

Prerequisites: Computer Organization, Data Structures

Course Objectives:

Familiarize students with

1. Basic functions and concepts of modern operating systems.
2. Mechanisms to handle processes and threads.
3. Principles of concurrency and deadlock.

Course Outcomes:

Students should be able to

1. Explain the structure of Operating System and basic architectural components.
2. Apply concepts of process, thread and scheduling.
3. Identify different memory management techniques.
4. Design solutions using IPC and Deadlock handling techniques.

Unit – I Introduction to Operating Systems (07)

Evolution of Operating Systems, Operating Systems Overview, OS structure, Functions of an OS: Program management, resource management, Protection and Security, PC Hardware and Booting, Shell Scripting, AWK, Sed

Unit – II Memory Management (07)

Logical Versus Physical Address Space, Swapping, Contiguous memory allocation, Non-contiguous memory allocation, Internal and external fragmentation, Segmentation, Paging, Structure of the Page Table

Virtual Memory: Demand paging, Prepaging, Thrashing, Page replacement algorithms, Translation look-aside buffer (TLB)

Unit – III Process (07)

Process concept, forking and exec, zombies, orphans, demons, context switching, wait, exit system calls, Scheduling: threads and scheduling algorithms, scheduling algorithms (FCFS, SJF, SRTF, Round robin, multilevel queues, feedback queues)

Linux schedulers -- O(1) and O(n), Linux schedulers – CFS

Unit – IV IPC and Synchronization (07)

IPC, Critical Section, Race Condition, context switching, process related system calls, Critical Sections, Peterson's Solution, Bakery Algorithm, Test&Set, Spinlocks, Mutex, semaphores, producer-consumer, dining philosophers

Deadlocks: Ostrich algorithm, bankers algorithm, deadlock prevention, deadlock detection and recovery

Unit – V I/O and File Management (07)

I/O Devices, Organization of the I/O Function, polling, Disk structure, Disk scheduling and Disk management, files, protection, access methods, directory and disk structure, File-system mounting, File-system structure and File-system implementation, allocation methods

Unit – VI System Software and its importance (07)

Need of System Software, Assemblers: Pass structure of Assemblers, Macro Processor: Macro Definition and call, Macro Expansion. Loaders: Loader Schemes, Compile and Go, General Loader Scheme, Subroutine Linkages, Relocation and linking

Text Books

1. “Operating System Concepts", 9th edition, by Abraham Silberschatz, Pert B. Galvin, and Greg Gagne, Wiley-India edition
2. "Modern Operating Systems", 4th edition, by Andrew S. Tanenbaum, PHI Learning Private Limited, New Delhi

Reference Books:

1. “Operating Systems: Internals and Design Principles”, 8th edition, William Stallings, Pearson Education Limited
2. “The Design of the UNIX Operating System”, Maurice J. Bach , Pearson
3. “UNIX, concepts and applications”, 4th edition, Sumitabha Das, Tata McGraw-Hill Education
4. "Operating Systems Security", Trent Jaeger, Morgan and Claypool Publishers
5. “Linux System Programming”, 2nd Edition, Robert Love, O’Reilly
6. “Systems Programming and Operating Systems”, 2nd Edition, D. M. Dhamdhere, Tata McGraw-Hill
7. “Systems Programming”, Indian Edition, J. J. Donovan, McGraw-Hill

IT 3203 Software Engineering

Teaching Scheme:
Lectures: **3** hrs/week
Tutorial: **1** hr/week

Examination Scheme:
In-Semester: **50** marks
End-Semester: **50** marks
Credits: **4**

Course Objectives:
Familiarize students with

1. Nature of software complexity in various application domains, disciplined way of software development and software lifecycle process models.
2. Concepts and principles of software design and architecture.
3. Basics of software testing through real life projects
4. Recent trends in software engineering.

Course Outcomes:
Students will be able to:

1. Identify unique features of various software application domains
2. Apply appropriate software development models for real life projects.
3. Identify functional and non-functional requirements for a small-to-medium size software project from real life projects
4. Examine the quality of the software

Unit – I Introduction to Software Engineering (07)
Nature of Software – How is software built? Software Application domains, web-apps, mobile-apps, cloud computing, Preliminaries – The discipline, layers, the process (guiding principles), the practice (guiding principles) and myths, Process Models – Generic process model, process assessment and improvement, prescriptive models, specialized models

Unit – II Software Requirement Analysis (07)
Requirements Capturing - requirements engineering (elicitation, specification, validation, negotiation, prioritizing requirements (kano diagram)
Requirements Analysis – basics, scenario based modeling, use case model, use case model development, data and control flow model, behavioral modeling using state diagrams
- real life application case study

Unit – III Software Design (07)
Software Design – definition of design, translating requirements model to design model, design considerations (quality guidelines and attributes), design concepts, Introduction to class identification, class relationships, identification of class relationships, Software architecture
UI Design - dealing with different types of users, collecting user-requirements, building narratives, creating personas and scenarios- real life application case study

Unit – IV Software Testing (07)
Software testing basics, Types of testing - unit testing and integrated testing, white box and black box testing, alpha and beta testing, regression testing, Peer testing, Art of debugging,
Software maintenance - real life application case study
Project quality management (CMMI, ISO, Six-sigma)

Unit – V Software Project Management (07)

Planning Scope Management, Creating the Work Breakdown Structure, Effort estimation and scheduling: Developing the Schedule using Gantt Charts, Using Tracking Gantt Charts to Compare Planned and Actual Dates, Critical Path Method, Program Evaluation and Review Technique (PERT) with examples. Estimating Costs, Types of Cost Estimates, Cost Estimation Tools and Techniques, Typical Problems with IT Cost Estimates, Introduction to automated Project management tools

Unit – VI Recent trends in Software Engineering (07)

Computer-aided software engineering (CASE), Risk Management, Software Configuration Management: Tools such as GitHub, Agile development process, Extreme Programming, SCRUM, Cleanroom methodology

Project management trends such as ERP, SAP, Global software development, Test-driven development

Text Books

Roger S. Pressman, '**Software Engineering: A practitioner's approach'**, *McGraw Hill*

Reference Books

1. Pankaj Jalote, '**An integrated approach to Software Engineering'**, *Springer/Narosa*.
2. Ian Sommerville, '**Software Engineering'**, *Addison-Wesley*.
3. Schwaber, K. and Beedle, M. (2001). '**Agile Software Development with SCRUM'**, *New Jersey:Pearson*. [ISBN - 9780130676344]
4. Rajiv Mall

PEIT 3201 Advanced Computer Networks

Teaching Scheme:
Lectures: **3** hrs/week

Examination Scheme:
In-Semester: **50** marks
End-Semester: **50** marks
Credits: **3**

Prerequisites: Foundations of Computer Networks, Computer Networks

Course Objectives:
Familiarize students with

1. Basic functions and concepts of advance computer networks.
2. Principles of performance modeling.
3. Mechanisms to handle congestion and routing.

Course Outcomes:
Students should be able to

1. Compare resource allocation mechanisms in computer networks.
2. Evaluate the performance measures in TCP/IP networks.
3. Analyze advanced routing algorithms.
4. Comprehend Internet design principles.

Unit – I Internet architecture and performance modeling (07)

Introduction. Course logistics. Goals of internet design, Layering abstraction and encapsulation. Network architecture and protocols. Performance of networks: delay and throughput, End-to-end delay, Concept of packetization, Circuit switching vs packet switching, Bandwidth-delay product, and Simple results from queuing theory.

Unit – II Applications: architectures and examples (07)

Application layer architectures: client-server vs. P2P, Socket interface: TCP vs. UDP semantics, Application types: elastic vs. real-time, WWW and HTTP. Persistent vs. non-persistent connections, HTTP message formats, headers, Caching, cookies, FTP, SMTP

Unit – III Transport protocols (07)

Basic function of transport - multiplexing and demultiplexing, UDP- simple transport, TCP connection basics: handshake, reliability, pipelining, congestion control, flow control, Ideal window size and bandwidth delay product, Buffer sizing for TCP, Simple model for TCP throughput, Understanding TCP fairness, RED gateways, Resource allocation, QoS, and fairness, QoS architectures: Intserv and Diffserv, Admission control: Token Bucket Filter

Unit – IV Internet routing (07)

Router scheduling, common router scheduling policies / queuing disciplines
Hierarchical (intradomain and interdomain) routing, IPv6, IP-in-IP tunneling, MPLS, BGP and advanced BGP concepts

Unit – V Link layer (07)

Link layer functions: Link layer addresses, ARP, Shared broadcast, multiple access protocols, the original Ethernet, spanning tree protocol, VLANs, NAT traversal.

Unit – VI Advanced topics (07)

Networking with virtual machines, software switches, Network Function Virtualization, Network Virtualization, Key ideas of traditional networks vs. SDN, history, Ethane: the motivation, OpenFlow: the interface, Onix: SDN controllers, Applications - B4 by Google, Datacenter networking.

Text Books

1. "Computer Networking, A Top-Down Approach", 6th edition, James Kurose and Keith Ross, Pearson Publishers.
2. "Computer Networks, A Systems Approach", 5th edition, Larry Peterson and Bruce Davie, The Morgan Kaufmann series in Networking
3. "Data Networks" 2nd edition Bertsekas and Gallager, Prentice hall publishers (mainly Chapter 3.3 on basic queuing theory

Reference Books

1. "Computer Networks", 6th Edition Andrew Tannenbaum, J. David and Wetherall, Pearson Publishers
2. Tcp/Ip Protocol Suite" 4th Edition Behrouz Forozoun, Tata McGraw-Hill Education Edition.

Reference Papers

1. The design philosophy of the DARPA internet protocols, David Clark.
2. Chord: A Scalable Peer-to-peer Lookup Service for Internet Applications, Stoica et al
3. Congestion Avoidance and Control, Jacobson and Karels.
4. Sizing Router Buffers, Appenzeller et al
5. Bufferbloat: Dark Buffers in the Internet, Gettys and Nichols
6. The Macroscopic Behavior of the TCP Congestion Avoidance Algorithm, Mathis et al.
7. Analysis of the Increase and Decrease Algorithms for Congestion Avoidance in Computer Networks, Chiu and Jain.
8. Random Early Detection Gateways for Congestion Avoidance, Floyd and Jacobson

PEIT 3201 Advanced Computer Architecture

Teaching Scheme:
Lectures: **3** hrs/week

Examination Scheme:
In-Semester: **50** marks
End-Semester: **50** marks
Credits: **3**

Prerequisites: Basics of Computer Architecture

Course Objectives:

Familiarize students with

1. The advanced architectural design of processors.
2. Computer architecture issues.
3. Various techniques to obtain performance improvement.

Course Outcomes:

Students should be able to

1. Comprehend new trends and developments in Computer Architecture
2. Evaluate performance of different architectures using various parameters
3. Analyze different Parallelism techniques
4. Evaluate cache and memory related issues in multi-processors

Unit – I Fundamentals Of Computer Design (07)

Review of Fundamentals of CPU, Memory and I/O – Technology Trends, power, energy and cost, Dependability, Performance Evaluation.

Unit – II Instruction Level Parallelism (07)

Basic concepts and challenges, Data dependencies and hazards , Overcoming data hazards, Dynamic Scheduling, reducing branch costs, high performance instruction delivery, hardware based speculation, limitation of ILP

Unit – III ILP With Software Approach (07)

Basic compiler techniques, Basic pipeline scheduling and loop unrolling, static branch prediction, Static multiple issue: VLIW approach, Hardware support for more ILP at compile time, Hardware versus Software Solutions.

Unit – IV Memory Hierarchy Design (07)

Introduction; Cache performance review, Cache performance, Average memory access time and processor performance, reducing cache miss penalty and miss rate, Memory Technology, Virtual memory

Unit – V Multiprocessors And Thread Level Parallelism (07)

Introduction, Challenges of parallel processing, Symmetric shared memory architectures, Multiprocessor cache coherency, distributed shared memory architecture, Directory based cache coherency Protocols, Synchronization, multi threading.

Unit – VI Storage Systems (07)

Introduction, I/o performance, CPU performance, Types of storage devices, Buses: connecting I/O devices to CPU /Memory, Redundant arrays of inexpensive Disks (RAID), errors and failures, I/O

performance Measures: throughput versus response time.

Text Books

1. John L Hennessey and David A Patterson, “Computer Architecture A Quantitative Approach”, Morgan Kaufmann/ Elsevier, Fifth Edition, 2012.
2. Kai Hwang and Faye Briggs, “Computer Architecture and Parallel Processing”, Mc Graw-Hill International Edition, 2000.

Reference Books

1. Sima D, Fountain T and Kacsuk P, “Advanced Computer Architectures: A Design Space Approach”, Addison Wesley, 2000.
2. David E. Culler, Jaswinder Pal Singh ,Parallel Computer Architecture, A Hardware / Software Approach, , Elsevier

PEIT 3201 Human Computer Interaction

Teaching Scheme:
Lectures: **3** hrs/week

Examination Scheme:
In-Semester: **50** marks
End-Semester: **50** marks
Credits: **3**

Prerequisites: Problem Solving and Object Oriented Technologies.

Course Objectives:

Familiarize students with

1. Basic field of human-computer-interaction study
2. Applications of human-computer-interaction to real life use cases.
3. Design of effective human-computer-interactions

Course Outcomes:

Students should be able to

1. Identify importance of HCI study and principles of User-Centered Design (UCD) approach.
2. Design effective user-interfaces following a structured and organized User Centered Design process.
3. Apply proper learning algorithm to data depending on the task
4. Perform evaluation of usability of a user-interface design.

Unit – I Introduction (07)

What is HCI? Disciplines involved in HCI, Why HCI study are important? The psychology of everyday things, Principles of HCI, User-centered Design.

Unit – II Understanding The Human (07)

Input-output channels, Human memory, Thinking: Reasoning and Problem Solving, Human emotions, Individual differences, Psychology and Design.

Unit – III Understanding The Interaction (07)

Models of interaction, Ergonomics, Interaction styles, WIMP Interface, Interactivity, Context of interaction, User experience, Paradigms of Interactions.

Unit – IV HCI - Design Process (07)

What is interaction design?, The software design process, User focus, Scenarios, Navigation Design, Screen Design, Prototyping techniques, Wire-Framing, Understanding the UI Layer and Its Execution Framework, Model-View-Controller(MVC) Framework.

Unit – V HCI - Design Rules , Guidelines And Evaluation Techniques (07)

Principles that support usability, Design standards, Design Guidelines, Golden rules and heuristics, Using toolkits, User interface management system (UIMS), Goals of evaluation, Evaluation Criteria, Evaluation through expert analysis, Evaluation through user participation, Choosing an Evaluation Method.

Unit – VI HCI Models And Theories (07)

Goal and task hierarchy model, Linguistic model, Physical and device models, Cognitive architectures, Hierarchical task analysis (HTA), Uses of task analysis, Diagrammatic dialog design

notations, Computer mediated communication, Ubiquitous Computing, Finding things on web Future of HCI.

Text Books

1. Alan Dix (2008). Human Computer Interaction. Pearson Education. ISBN 978-81-317-1703-5.
2. Gerard Jounghyun Kim (20 March 2015). Human–Computer Interaction: Fundamentals and Practice. CRC Press. ISBN 978-1-4822-3390-2.

Reference Books

1. Ben Shneiderman; Catherine Plaisant; Maxine Cohen; Steven Jacobs (29 August 2013). Designing the User Interface: Strategies for Effective Human-Computer Interaction. Pearson Education Limited. ISBN 978-1-292-03701-1.
2. Donald A. Norman (2013). The Design of Everyday Things Basic Books. ISBN 978-0-465-07299-6.
3. Jeff Johnson (17 December 2013). Designing with the Mind in Mind: Simple Guide to Understanding User Interface Design Guidelines. Elsevier. ISBN 978-0-12-411556-9.
4. Alan Cooper; Robert Reimann; David Cronin; Christopher Noessel (13 August 2014). About Face: The Essentials of Interaction Design. Wiley. ISBN 978-1-118-76658-3.

PEIT 3202 Multimedia Computing

Teaching Scheme:
Lectures: **3** hrs/week

Examination Scheme:
In-Semester: **50** marks
End-Semester: **50** marks
Credits: **3**

Prerequisites: Algebra and Geometry

Course Objectives:

Familiarize students with

1. Variety of multimedia data modification algorithms
2. Capturing and using multimedia components for presenting a concept
3. Moderate level of multimedia data processing for its size reduction
4. Usage of multimedia in variety of domain applications

Course Outcomes:

Students should be able to

1. Know the purpose of multimedia components in multimedia production.
2. Apply data processing techniques on multimedia data
3. Apply compression techniques on multimedia data
4. Choose different multimedia components for multimedia system design

Unit – I Multimedia Overview and basics of still Image (07)

Multimedia Overview: Introduction, multimedia presentation and production, characteristics of multimedia presentation, hardware and software requirements, uses of multimedia, analog and digital representation, digitization, Nyquist theorem, quantization error, visual display systems.

Digital Image: Image as data, Image acquisition, types of images

Unit – II Image Processing (07)

Binary image processing, grey scale image processing, colored image processing. Image output on monitors, image output on printers, image file formats both lossless and lossy

Unit – III Audio data as multimedia component (07)

Introduction, acoustics, sound waves, types and properties of sound, psycho acoustics, components of an audio system, digital audio, synthesizers, MIDI, audio processing

Unit – IV Audio transmission and broadcasting (07)

Speech, sound card, audio transmission, digital audio broadcasting, surround sound system, audio file formats both lossless and lossy

Unit – V Video data as multimedia component (07)

Motion video, digital video, digital video processing, video recording and storage formats both lossless and lossy, video editing concepts

Unit – VI Data compression (07)

Image compression, audio compression, video compression in all six techniques to be studied

Text Books

1. Rajan Parekh: Principles of multimedia, TMH 2nd Edition-2013.
2. Nigel Chapman, Jenny Chapman Peter: Digital Multimedia, John Wiley and sons, Edition 2012.

Reference Books

1. Kimberly N Rosenfield: Digital Multimedia: Concepts, Methodologies, tools and applications, Information Resources Management association 1st Edition-September 2017

PEIT 3202 Natural Language Processing

Teaching Scheme:
Lectures: **3** hrs/week

Examination Scheme:
In-Semester: **50** marks
End-Semester: **50** marks
Credits: **3**

Prerequisites: Probability Basics

Course Objectives:

Familiarize students with

1. Different levels of natural language processing
2. Language modeling and Parsing techniques used in natural language processing
3. State of art NLP areas

Course Outcomes:

Students will be able to:

1. Identify challenges involved in developing natural language processing system
2. Analyze natural language processing techniques
3. Choose Natural Language Processing techniques for different applications
4. Evaluate natural language processing system

Unit – I Introduction to Natural Language Processing (07)

Introduction: What is Natural Language Processing? Introduction to NLP applications, Brief history of field, Ambiguity and Uncertainty in language, The Different Levels of Language Analysis : NLP tasks in syntax, semantics and pragmatics, The role of machine learning

Unit – II Syntactic Parsing (07)

A Top-Down Parser, A Bottom-Up Chart Parser, Top-Down Chart Parsing, Logic Programming Parsing tools such as Stanford Parser, Human Preferences in Parsing, Application of Natural Language toolkit

Unit – III Language Modeling (07)

Computational Linguistics - Probability Theory , Estimating Probabilities, Part-of-Speech Tagging, Obtaining Lexical Probabilities, Probabilistic Context-Free Grammars, Probabilistic language modeling and its applications, Markov models. N-grams. Estimating the probability of a word, and smoothing

Unit – IV Features and Augmented Grammars (07)

Feature Systems and Augmented Grammars : Some Basic Feature Systemsfor English Morphological Analysis and the Lexicon , A Simple Grammar Using Features, Parsing with Features, Augmented Transition Networks: Definite Clause Grammars, Generalized Feature Systems and Unification Grammars

Unit – V Semantic Analysis (07)

Semantics and Logical Form :Word Senses and Ambiguity,The Basic Logical Form,Language Encoding, Ambiguity in Logical Form ,Verbs and States in Logical Form,Case Relations Lexical Resources : WordNet, Semantic web Ontologies

Unit – VI Future of NLP (07)

Sentiment Analysis. Machine Translation MT evaluation tools such as Bleu, WER, Information Extraction, Question answering, Automatic speech recognition, Deep Learning for Natural Language Processing

Text Books

1. James Allen, “Natural Language Understanding”, Pearson Publication, ISBN: 978-81-317-0895-8 2nd Edition
2. D. Jurafsky, J. H. Martin, “Speech and Language Processing”, Pearson Education, 2002

Reference Books

1. Christopher D. Manning, Hinrich Schutze, Foundations of Statistical Natural Language Processing, The MIT Press, Cambridge, Massachusetts, 1999.
2. Tanveer Siddiqui, US Tiwary, Natural Language Processing and Information Retrieval
3. Daniel M.Bikel, ImedZitouni, Multilingual Natural Language Processing Applications

PEIT 3202 Advanced Machine Learning

Teaching Scheme:
Lectures: **3** hrs/week

Examination Scheme:
In-Semester: **50** marks
End-Semester: **50** marks
Credits: **3**

Prerequisites: Foundations of Machine Learning, Probability Basics

Course Objectives:

Familiarize students with

1. Selection of appropriate features for processing
2. Advanced concepts of Machine Learning
3. Applications of advanced concepts of Machine Learning

Course Outcomes:

Students should be able to

1. Group similar types of tasks
2. Apply advanced machine learning techniques to variety of applications
3. Compare various algorithms
4. Recommend learning technique depending upon the task

Unit – I Dimensionality Reduction (07)

Introduction to Dimensionality Reduction, Feature Selection, Subset Selection, Principal Component Analysis, Linear Discriminant Analysis

Unit – II Ensemble and Reinforcement Learning (07)

Ensemble Learning – Combining Multiple Models, Bagging, Randomization, Boosting, Additive Regression

Reinforcement Learning – Introduction, Single State Case: K-armed Bandit, Elements of Reinforcement Learning, Reward, Penalty, Exploration and Exploitation

Unit – III Artificial Neural Network (07)

Biological neuron, Artificial neuron, Pattern recognition tasks, Features of artificial neural networks, Activation functions, learning parameters, Perceptron, single layer perceptron, Architectures of multilayer feed-forward neural networks, Error Back-propagation learning algorithm, Generalized delta rule, Limitations of MLP

Unit – IV Convolution Neural Network (07)

Invariance, stability, Variability models (deformation model, stochastic model), Scattering networks

Group Formalism, Supervised Learning: classification, Properties of CNN representations: inevitability, stability, invariance. Covariance /invariance: capsules and related models. Connections with other models: dictionary learning, LISTA. Other tasks: localization, regression

Unit – V Evolutionary Computing and Swarm Intelligence (07)

Biological background, Overview of evolutionary computing, Genetic Algorithm and Search Space, Operators in Genetic Algorithm – encoding, selection, crossover and mutation, classification of Genetic Algorithms

Swarm Intelligence and Ant Colony Optimization

Unit – VI Fuzzy Systems (07)

Basic Fuzzy logic theory, sets and their properties, Operations on fuzzy sets, Fuzzy relation and operations on fuzzy relations and extension principle, Fuzzy membership functions and linguistic variables, Fuzzy rules and fuzzy reasoning, Fuzzification and defuzzification and their methods, Fuzzy inference systems and Fuzzy knowledge based controllers

Text Books

1. Ethem Alpaydin: Introduction to Machine Learning, PHI 2nd Edition-2013.
2. S. Haykin: Neural Networks: A Comprehensive Foundation, 2nd Edition, Pearson Education Asia, 1999

Reference Books

1. Ian H Witten, Eibe Frank, Mark A Hall: Data Mining, Practical Machine Learning Tools and Techniques, Elsevier, 3rd Edition.
2. S. N. Sivanandam. S. Sumathi and S. N. Deepa: Introduction to Neural Network using MATLAB, Tata McGraw-Hill Publications
3. B. Kosko: Neural Networks and Fuzzy Systems – A Dynamical Systems Approach to Machine Intelligence, Prentice-Hall of India, 1994
4. Jack > Zurada: Introduction to Artificial Neural Systems, PWS Publishing Co, Boston, 2002
5. David E. Goldberg: Genetic Algorithms, Pearson Education, 2006
6. Parag Kulkarni: Reinforcement Learning and Systemic Machine Learning for Decision Making, IEEE Press, Reprint 2015.

PEIT 3202 Systems Programming

Teaching Scheme:
Lectures: **3** hrs/week

Examination Scheme:
In-Semester: **50** marks
End-Semester: **50** marks
Credits: 3

Prerequisites: Data Structures & Files, Theory of Computation

Course Objectives:

Familiarize students with

1. Architecture of assembler, macro processor, linker & loader
2. Construction of compiler
3. Code optimization techniques

Course Outcomes:

Students should be able to

1. Design a two pass assembler for a given hypothetical assembly language program.
2. Design a two pass macro preprocessor for a given hypothetical assembly language program.
3. Build syntax tree & annotated parse tree for a given code statement.
4. Apply different code optimization techniques.

Unit – I Introduction to System Programming & Assembler (07)

Introduction: Need of System Software, Components of System Software, Language Processing Activities, Fundamentals of Language Processing.

Assemblers: Elements of Assembly Language Programming, A simple Assembly Scheme, Pass structure of Assemblers, Design of Two Pass Assembler, Single pass assembler.

Unit – II Macro processor, Linker & Loader (07)

Macro processor: Design of two-pass Macro Processor, nested macro definition

Linker & Loader: General Loader Scheme, Absolute Loader Scheme, Relocation and linking concepts, Relocating Loaders, Direct Linking Loaders, Overlay Structure.

Unit – III Introduction to Compiler & Lexical Analysis (07)

Introduction: Phases of compiler, compilation process

Lexical analysis: Role of lexical analysis, Design of Lexical Analyzer using Uniform Symbol Table, lexical errors. Generation of Lexical Analyzer by LEX.

Unit – IV Syntax Analysis (07)

Introduction: Role of syntax analysis, Top-down, Bottom-up, Predictive, Table driven parser

Top-down parser: Recursive descent parser – generation, use

Bottom-up parser: SLR, LALR parser – generation, use

YACC specification and Automatic construction of Parser (YACC).

Unit – V Semantic Analysis & Intermediate Code Generation (07)

Semantic analysis: Parse tree, syntax tree, annotated parse tree, Type checking & type conversion, Syntax direction translation for assignment, conditional & iterative statements, construction of syntax

tree & directed acyclic graph
Intermediate code generation: Postfix, Three address code – quadruple, triple, indirect triple

Unit – VI Code optimization & generation (07)
Code optimization: Machine Independent: Peephole optimizations: Common Sub-expression elimination, Removing of loop invariants, Induction variables and Reduction in strengths, use of machine idioms
Code generation: Sethi Ulman algorithm for code generation

Text Books

1. D. M. Dhamdhere, Systems Programming and Operating Systems, Tata McGraw-Hill, ISBN 13:978-0-07-463579-7, Second Revised Edition.
2. Alfred V. Aho, Ravi Sethi, Jeffrey D. Ullman, Compilers Principles, Techniques and Tools, Addison Wesley, ISBN:981–235–885 - 4, Low Price Edition.

Reference Books

1. J. J. Donovan, Systems Programming, McGraw-Hill, ISBN 13:978-0-07-460482-3, Indian Edition.
2. Leland L. Beck, System Software An introduction to Systems Programming, Pearson Education, ISBN13: 9788177585551.
3. John R. Levine, Linkers & Loaders – Harcourt India Pvt. Ltd., Morgan Kaufmann Publishers, 2000, ISBN 13: 978-1558604964.

IT 3204 Seminar

Teaching Scheme:
Practical: **2** hrs/Week

Examination Scheme:
Oral: **25** Marks

Credits: 1

Course Objectives:

Familiarize students with

1. Exploring technical literature with the purpose of formulating a project statement.
2. Writing a technical report summarizing state-of-the-art on an identified topic.
3. Formulate intended future work based on the technical review.
4. Understanding scientific approach for literature survey and paper writing.
5. Developing a prototype for the project statement.

Course Outcomes:

Students should be able to

1. Perform focused study of technical literature relevant to a specific topic.
2. Build independent thinking abilities to approach complex problems.
3. Work as a team and follow collaborative work practices.
4. Communicate scientific information to a larger audience in oral and written form.
5. Develop prototype to test and validate project statement

Guidelines for Project Based Seminars

1. A project group consisting of 4 students shall identify problem(s) in Computer Engineering / Information Technology referring to recent trends and developments in consultation with institute guide.
2. Students can choose problem statements from websites with international competitions/ challenges like kaggle, hackerrank, hackerone, bug bounty, bugcrowd etc. and attempt to solve current challenges/ unsolved problems preferably.
3. The group must review sufficient literature (reference books, journal articles, conference papers, white papers, magazines, web resources etc.) in relevant area on their project topic as decided by the guide.
4. Students should not perform the similar projects/ assignments done before.
5. Students should develop some working prototype/ Hands on development/ participate in competitions etc.
6. The topic and scope should be verified and approved by the guide.
7. Project Statement verification will be done by group of faculty members.
8. Individual seminar topics will be discussed and finalized by the guide.

Guidelines for Seminar Report

1. Each student shall submit two copies of the seminar report in a prescribed format duly signed by the guide and Head of the department/Principal.
2. First chapter of a project group may talk about the project topic. At the end of the first chapter individual students should begin with introduction of seminar topic and its objectives.
3. Broad contents of review report (20-25 pages) shall be
    i. Introduction of Project Topic
    ii. Motivation, purpose and scope of project and seminar
    iii. Related work (of the seminar title) with citations
    iv. Discussion (your own reflections and analysis)
    v. Conclusions
    vi. Project definition
    vii. References in IEEE Format
4. Students are expected to use open source tools for writing seminar report, citing the references and plagiarism detection. (Latex, for report writing ; Mendeley, Zotero for collecting, organizing and citing the resources; DupliChecker , PaperRater, PlagiarismChecker, Turnitin and Viper for plagiarism detection)

Guidelines for Seminar Evaluation

1. A panel of examiners - one External examiner & one internal examiner (other than guide) will assess the seminar during the presentation.
2. Criteria for evaluation
    a) New technology / topics.
    b) Working of the prototype developed / Experimentation.
    c) Literature survey
    d) Presentation skills
    e) Utility of project

Project based seminar Timeline

1.	Group formation	1st week
2.	Survey / Topic identification	3rd week
3.	Topic verification (Refer guideline for project based seminar no 8)	4th week
4.	Guide allocation	5th week
5.	Seminar topics identification	6th week
6.	Prototype development/ hands-on	10th week
7.	Seminar report submission	11th week
8.	Presentation	12th week

References

1. Sharon J. Gerson, Steven M. Gerson, Technical Writing: Process and Product, Pearson Education Asia, ISBN :130981745, 4th Edition.
2. Andrea J. Rutherfoord, Basic Communication Skills for Technology, Pearson Education Asia, 2nd Edition.
3. Lesikar, Lesikar's Basic Business Communication, Tata McGraw, ISBN :256083274, 1st Ed.

IT 3205 Programming Skills Development Laboratory

Teaching Scheme:
Practical: **2** hrs/week

Examination Scheme:
In-Semester: **25** Marks
Credits: **1**

Prerequisites: Object Oriented Programming.

Course Objectives:
Familiarize students with

1. Learn android app development and Java programming
2. The Google app store management
3. Successfully design, code and deploy Android apps.

Course Outcomes:
Students should be able to

1. Learn Android development, Java programming and Android studio from scratch.
2. Breaks even the most complex applications down into simplistic steps.
3. Learn how to work with APIs, web services and advanced databases.
4. Upload your android apps to the Google play and reach millions of android users

Course Contents

1. Set up and walkthrough - Android Studio and build User Interface.
2. Fundamentals of Java Programming used to build Android apps.
3. Inputs, Buttons and Reactive Interfaces.
4. Android Building blocks.
5. Variables, Arrays, Loops, Array Lists, List View.
6. Navigate between screens.
7. Passing information between screens.
8. Think and work - Learn how professional android apps developers.
9. Think and work - Learn how to design android apps.
10. Build several amazing apps - Hands on.
11. Publish your apps on Google Play.

List of Lab Experiments

1. Develop an application that uses GUI components, Fonts and Colors
2. Develop an application that uses Layout Managers and event listeners.
3. Develop a native calculator application.
4. Write an application that draws basic graphical primitives on the screen.
5. Develop an application that makes use of database.
6. Develop an application that makes use of RSS Feed.
7. Mini Project – Design and Develop of Android App with bit of societal angle. App to be deployed on Android Play Store

Text Books

1. Hello, Android: Introducing Google's Mobile Development Platform (Pragmatic Programmers) 3rd Edition by Ed Burnette, Paperback

2. Android Application Development All-in-One for Dummies 1st Edition, Barry A. Burd. Paperback

Reference Books

1. Beginning Android Tablet Application Development 1st Editionby Wei-Meng Lee, Paperback
2. Professional Android 2 Application Development 2nd Edition,Reto Meier, Paperback.
3. Android Programming for Beginners by John Horton Paperback.

IT 3206 Operating Systems Laboratory

Teaching Scheme:
Practical: **4** hrs/week

Examination Scheme:
In-Semester: **25** marks
Practical: **25** marks
Credits: **2**

Prerequisites: Computer Organization, Data Structures

Course Objectives:

Familiarize students with

1. Shell scripting and its importance.
2. Concepts of processes and threads.
3. Concurrency, Synchronization and deadlocks.
4. Basics of Linux Kernel Programming.

Course Outcomes:

Students should be able to

1. Examine the importance and functioning of shell programming.
2. Illustrate the benefits of thread over process and implement synchronized programs using multithreading concepts.
3. Analyse the concept of deadlock in operating systems and implement it in multiprocessing environment.
4. Design solutions using mutual exclusion, IPC and synchronization.

Suggested List of Laboratory Assignments

1. Create two virtual machines using Type-2 hypervisor having Fedora and FreeBSD installed on them, to understand basic virtualization concept.
2. Shell programming.
3. Write C programs to simulate UNIX commands like ls, grep, etc.
4. Write programs using the following system calls of UNIX operating system: fork, exec, getpid, exit, wait, close, stat, opendir, readdir
5. Write a C program to implement multithreading.
6. Implement producer-consumer problem using semaphores.
7. Write a C program to simulate the concept of Deadlock using Dining-Philosophers problem.
8. Write a C program to implement Inter Process Communication (shared memory or pipes or message queues).
9. Write a C program that uses polling to simultaneously check whether a read from stdin and a write to stdout will block.
10. Build and insert loadable kernel module to a running Linux kernel.

Reference Books:

1. “Beginning Linux Programming”, 4th Edition, by Neil Matthew, Richard Stones, Wrox Publication
2. “UNIX, concepts and applications”, 4th Edition, Sumitabha Das, Tata McGraw-Hill Education
3. “Linux System Programming”, 2nd Edition, Robert Love, O’Reilly
4. “Linux Kernel Development”, 3rd Edition, Robert Love, Pearson
5. “Operating System Concepts", 9th edition, Abraham Silberschatz, Pert B. Galvin, and Greg Gagne, Wiley-India edition
6. "Modern Operating Systems", 4th edition, Andrew S. Tanenbaum, PHI Learning Private Limited, New Delhi
7. “Operating Systems: Internals and Design Principles”, 8th edition, William Stallings, Pearson Education Limited

PEIT 3203 Multimedia Computing Laboratory

Teaching Scheme:
Practical: **2** hrs/week

Examination Scheme:
Practical: **25** marks
Credits: **1**

Prerequisites: Data Structures, Algorithms, Geometry, Trigonometry, Vectors and Matrices

Course Objectives:
Familiarize students with
1. Use of multimedia libraries for data processing

Course Outcomes:
Students should be able to
1. Apply filters and highlight some part of the multimedia data, based on the application
2. Develop programs for manipulating multimedia data.
3. Use different multimedia algorithms.
4. Choose algorithms based on the multimedia application.

List of Assignments

1 Familiarity with the platform for image and audio processing python or Java Advanced Imaging (JAI), Java Binding on Open GL(JOGL), Java Media Framework (JMF) OPENIMAJ
2 Image processing spacial domain (Mean filter)
3 Image processing spacial domain (Gausian Filter)
4 Image processing frequency domain 1(Mean filter)
5 Image processing frequency domain 2 (Gaussian Filter)
6 Audio processing temporal domain (Normalization)
7 Audio Processing temporal domain/ Masking
8 Video Processing -1(Edge Detection)
9 Video Processing -2 (finding faces)
10 Multiple ways data processing on multimedia Data

PEIT 3203 Natural Language Processing Laboratory

Teaching Scheme:
Practical: **2** hrs/week

Examination Scheme:
Practical: **25** marks
Credits: **1**

Course Objectives:
Familiarize students with

1. Implementation of NLP core areas.
2. Current NLP research areas.

Course Outcomes:
Students will be able to

1. Apply various NLP algorithms and tools.
2. Analyze challenges in development of NLP applications.
3. Implement NLP application.
4. Design Language Model.

Assignments

1: Parsing using tools such as Natural Language toolkit (NLTK), Stanford Parser
2: Implementation of probabilistic context free grammar
3: Use of lexical resources to implement word sense disambiguation
4: Study of any small application/research paper in areas such as sentiment analysis, machine translation

PEIT 3203 Advanced Machine Learning Laboratory

Teaching Scheme:
Practical: **2** hrs/week

Examination Scheme:
Practical: **25** marks
Credits: **1**

Prerequisites: Linear Algebra and Calculus, Probability Basics, DBMS

Course Objectives:

Familiarize students with

1. Dimensionality reductions techniques
2. Various advanced machine learning techniques

Course Outcomes:

Students should be able to

1. Prepare data for learning and analytics
2. Apply advanced machine learning algorithms
3. Compare various algorithms
4. Use large datasets.

Suggested List of Laboratory Assignments

Implementation of programs to be done in Python

1. Apply PCA for Dimensionality Reduction
2. Classify data using Ensemble Learning algorithm
3. Recognize characters using Artificial Neural Network Technique

Text Books

1. Andreas Muller and Sarah Guido: Introduction to Machine Learning with Python, O'Reilly, 2017

Reference Books

1. Michael Bowles: Machine Learning in Python, Wiley, 2018
2. Ethem Alpaydin: Introduction to Machine Learning, PHI 2nd Edition-2013.

PEIT 3203 Systems Programming Laboratory

Teaching Scheme:
Practical: **2** hrs/week

Examination Scheme:
Practical: **25** marks
Credits: 1

Prerequisites: C programming, Data Structures & Files

Course Objectives:

Familiarize students with

1. Concepts of assembler to design and implement two pass assembler.
2. Lexical analyzer and parser and applications in compiler design.
3. Developing & using library files

Course Outcomes:

Students should be able to

1. Implement a two pass assembler for a given hypothetical assembly language program.
2. Implement a two pass macro preprocessor for a given hypothetical assembly language program.
3. Implement lexical and syntax analyzer for a subset of given C code snippet.
4. Develop static & dynamic link libraries.

List of Assignments (Any 5)

1. Write a C program to implement Pass-I of Two-pass assembler for Symbols and Literal processing (For hypothetical instruction set from Dhamdhere)
   considering following cases Forward references
   a. DS and DC statement
   b. START, EQU, LTORG, END.
   c. Error handling: symbol used but not defined, invalid instruction/register etc.
2. Write a C program to implement Pass-II of Two-pass assembler for output of previous assignment.
3. Study Assignment for Macro Processor.
4. Write a C program to implement Lexical Analyzer for subset of C.
5. Write a C program to implement a Recursive Descent Parser.
6. Write a program to implement calculator using LEX and YACC.
7. Write a program to create Static Link Library (.a) Dynamic Link Library (.so) for any mathematical operation and write an application program to test it.

Text Books

1. D. M. Dhamdhere, Systems Programming and Operating Systems, Tata McGraw-Hill, ISBN 13:978-0-07-463579-7, Second Revised Edition.
2. John Levine, Lex & Yacc, O'Reilly, ISBN-13: 978-1565920002

Autonomous Program Structure of
Final Year B. Tech. (Information Technology)
Academic Year: 2019-2020 Onwards

Final Year B. Tech. (IT) Semester – I										
Course Code	Course Title	Teaching Scheme Hours /Week			Examination Scheme				Marks	Credit
		Lecture	Tutorial	Practical	In Semester	End Semester	Oral	Practical		
IT 4101	Software Architecture & Design Patterns	3	0	0	50	50	0	0	100	3
IT 4102	Cloud Computing	3	0	0	50	50	0	0	100	3
HS 4101	Green Computing*	3	0	0	50	50	0	0	100	3
OE 4101	Open Elective –I	3	0	0	50	50	0	0	100	3
IT 4103	Software Architecture & Design Patterns Laboratory	0	0	2	0	0	50	0	50	1
IT 4104	Project Phase-I	0	2	14	100	0	50	0	150	9
	Total	**12**	**2**	**16**	**300**	**200**	**100**	**0**	**600**	**22**
	Grand Total	**30**			**600**				**600**	**22**

*Advanced Entrepreneurship Development**

**Prerequisite: Basic Course ED

MKSSS's Cummins College of Engineering for Women Karvenagar, Pune-411052

DEAN ACADEMICS
MKSSS's Cummins College of Engineering for Women Karvenagar, Pune-411052

Principal
MKSSS's Cummins College of Engg. For Women, Karvenagar, Pune-52.

APPROVED BY
Governing Body Members
MKSSS's Cummins College of Engineering for Women Karvenagar, Pune-411052

IT 4101 Software Architecture and Design Patterns

Teaching Scheme:
Lectures: **3** hrs/week
Tutorial: NIL

Examination Scheme:
In-Semester: **50** marks
End-Semester: **50** marks
Credits: **3**

Prerequisites: Object Oriented Paradigms, Software Engineering

Course Objectives:

Familiarize students with

1. Concepts of software architecture
2. Different types of software architectural styles
3. Concepts and applications of design patterns.
4. Different types of design patterns

Course Outcomes:

Students should be able to

1. Analyze and suggest architecture design for an application
2. Apply design patterns to software design
3. Evaluate and select appropriate design pattern for a situation
4. Compare the performance of the software on inclusion of various design patterns.

Unit – I: Software Architecture **(07)**

Overview of software Architecture, What drives software architecture, Quality attributes, Architecture design, Architecture documentation

Unit – II: Architectural Patterns **(07)**

Client server multitier architectural pattern, Even driven architectural pattern, Service Oriented Architectures, Component based architecture

Unit – III: Role of design patterns in architecture design **(07)**

Introduction to architecture design, introduction to design patterns, Types of design patterns Abstract factory, builder, factory method, singleton design patterns

Unit – IV: Creational Design Patterns **(07)**

Abstract factory, builder, factory method, singleton design patterns with case study

Unit – V: Structural Design Patterns (07)

Adapter, bridge, composite, facade, decorator, chain of responsibility with case study

Unit – VI: Behavioral Design Patterns (07)

State, Observer, Strategy, template method with case study,

Text Books:

1. Craig Larman, Applying UML and Patterns, Pearson Education, Second Edition, ISBN: 9780130925695.
2. Elizabeth Freeman, Kathy Seirra, Head first design patterns O'Reilly Media ISBN 0596007124

Reference Books:

1. Len Bass, Paul Clements, Rick Kazman Software Architecture in Practice, Pearson Education, ISBN: 978-81-7758-996-2
2. Eric Gamma and other authors Design Patterns Elements of reusable object oriented software Addison Wesley Professional Series ISBN 0-202-63361-2

IT 4102 Cloud Computing

Teaching Scheme:
Lectures: **3** hours/week
Tutorial: NIL

Examination Scheme:
In-Semester: **50** marks
End-Semester: **50** marks
Credits: **3**

Prerequisites: Operating Systems and Computer Networks

Course Objectives:
Familiarize students with
1. Distributed Systems and its ecosystem.
2. Basics of virtualization and its importance.
3. In-depth analysis of cloud computing capabilities.
4. Overview of cloud programming and services.

Course Outcomes:
Students should be able to
1. Recognize need of cloud based solutions.
2. Justify the importance of distributed systems.
3. Determine effective techniques to program cloud systems.
4. Evaluate current challenges and trade-offs in cloud computing.

Unit – I Introduction to Distributed Systems (07)
Scalable Computing over the Internet, Technologies for Network-Based Systems, System Models for Distributed and Cloud Computing, Software Environments for Distributed Systems and Clouds, Performance, Security, and Energy Efficiency

Unit – II Computer Clusters for Scalable Parallel Computing (07)
Clustering for Massive Parallelism, Computer Clusters and MPP Architectures, Design Principles of Computer Clusters, Cluster Job and Resource Management, Case Study: Top Supercomputer Systems

Unit – III Virtual Machines and Virtualization of Clusters and Data Centers (07)
Implementation Levels of Virtualization, Virtualization Structures/Tools and Mechanisms, Virtualization of CPU, Memory, and I/O Devices, Virtual Clusters and Resource Management, Virtualization for Data-Center Automation

Unit – IV Cloud Platform Architecture over Virtualized Data Centers (07)
Cloud Computing and Service Models, Data-Center Design and Interconnection Networks, Architectural Design of Compute and Storage Clouds, Public Cloud Platforms: GAE, AWS, and Azure, Inter-cloud Resource Management, Cloud Security and Trust Management

Unit – V Cloud Programming and Software Environments (07)

Features of Cloud and Grid Platforms, Parallel and Distributed Programming Paradigms, Programming Support of Google App Engine, Programming on Amazon AWS and Microsoft Azure, Emerging Cloud Software Environments,

Unit – VI Grids, P2P, and the Future Internet (07)

Grid Architecture and Service Modeling, Grid Projects and Grid Systems Built, Peer-to-Peer Computing Systems, Cloud Trends in Supporting Ubiquitous Computing, Enabling Technologies for the Internet of Things

Text Books

1. Jack J. Dongarra, Kai Hwang, Geoffrey C. Fox, Distributed and Cloud Computing: From Parallel Processing to the Internet of Things, Elsevier, First Edition

Reference Books

1. Thomas Erl, Zaigham Mahmood and Ricardo Puttini, Cloud Computing: Concepts, Technology & Architecture, Pearson, First Edition
2. Rajkumar Buyya, Christian Vecchiola, S.Thamarai Selvi, Mastering Cloud Computing: Foundations and Applications Programming, McGraw Hill, First Edition
3. A. Srinivasan, J. Suresh, Cloud Computing: A practical approach for learning and implementation, Pearson, First Edition
4. Anthony T. Velte, Cloud Computing: Practical Approach, McGraw Hill, and First Edition
5. Ronald L. Krutz and Russell D. Vines, Cloud Security: A Comprehensive guide to Secure Cloud Computing, Wiley, First Edition

HS 4101 Green Computing

Teaching Scheme:
Lectures: **3** hrs/week
Tutorial: NIL

Examination Scheme:
In-Semester: **50** marks
End-Semester: **50** marks
Credits: **3**

Prerequisites: Basic Sciences

Course Objectives:

Familiarize students with

1. Knowledge of green computing practices to minimize negative impacts on the environment.
2. Principles of green computing.
3. Green Computing and how it can help improve environmental sustainability.
4. Green Computing in enterprises and its impact.

Course Outcomes:

Students should be able to

1. Relate to the socio cultural aspects of green computing.
2. Create awareness about green computing and promote green agenda in their working environments leading to green movement.
3. Apply green computing skills such as energy efficiency, IT assets disposal, carbon footprint estimation, reporting and development of green products.
4. Justify green initiatives while developing applications and services in enterprises.

Unit – I: Introduction to Green Computing (07)

Environmental Impacts of IT, Need of green computing, Green IT Standards, Enterprise Green IT Strategy, Hardware: Reuse, Recycle and Dispose, present scenario in industry, health issues relevance, Software: Energy-Saving Software Techniques

Unit – II: Software Development and Green Data Centers (07)

Sustainable Software, Software Sustainability Attributes, Software Sustainability Metrics, Sustainable Software Methodology, Data Centres and Associated Energy Challenges, Data Centre IT Infrastructure, Data Centre Facility Infrastructure: Implications for Energy Efficiency, IT Infrastructure Management, Green Data Centre Metrics

Unit – III: Green Data Storage and Networks (07)

Storage Media Power Characteristics, Energy Management Techniques for Hard Disks, System-Level Energy Management, Objectives of Green Network Protocols, Green Network Protocols and Standards

Unit – IV: Enterprise Green IT Strategy (07)

Approaching Green IT Strategies, Business Drivers of Green IT Strategy, Business Dimensions for Green IT Transformation, Multilevel Sustainable Information,

Sustainability Hierarchy Models, Product Level Information, Individual Level Information, Functional Level Information, Organizational Level Information, Regional/City Level Information

Unit – V: Green Computing Services and Roles (07)

Factors Driving the Development of Sustainable IT, Sustainable IT Services (SITS), Sustainable IT Roadmap, Organizational and Enterprise Greening, Information Systems in Greening Enterprises, Greening the Enterprise

Unit – VI: Regulating Green Computing (07)

The Regulatory Environment and IT Manufacturers, Nonregulatory Government Initiatives, Industry Associations and Standards Bodies, Green Building Standards, Green Data Centres, Social Movements

Text Books:

1. San Murugesan, G. R. Gangadharan: Harnessing Green IT, WILEY, 1st Edition-2013.

Reference Books:

1. Woody Leonhard, Katherrine Murray, "Green Home computing for dummies", August 2009, WILEY
2. Bhuvan Unhelkar, "Green IT Strategies and Applications-Using Environmental Intelligence", CRC Press, June 2011
3. Alin Gales, Michael Schaefer, Mike Ebbers, "Green Data Center: steps for the Journey", Shroff/IBM redbook, 2011.
4. Jason Harris, "Green Computing and Green IT-Best Practices on regulations & industry", Lulu.com, 2008
5. Carl Speshocky, "Empowering Green Initiatives with IT", John Wiley & Sons, 2010.
6. Wu Chun Feng (Editor), "Green computing: Large Scale energy efficiency", CRC Press, 2012.

OE 4101 Software Testing and Quality Assurance

Teaching Scheme:
Lectures: **3** hrs/week
Tutorial: NIL

Examination Scheme:
In-Semester: **50** marks
End-Semester: **50** marks
Credits: **3**

Prerequisites: Any programming language

Course Objectives:

Familiarize students with

1. Application of testing strategies in projects.
2. Test management strategies and tools for testing
3. Various quality assurance models

Course Outcomes:

Students should be able to

1. Analyze the project scenario and to select proper testing technique
2. Apply testing techniques to deliver a project
3. Choose quality assurance models for the project
4. Choose one of the latest testing tools suitable for the project

Unit – I: Software testing fundamentals **(07)**

Testing as a Process, Software testing principles, The tester's role in a software development organization, Origins of defects, Defect classes, Testing fundamentals, the defect repository and test design, Defect examples, Developer /Tester support for developing a defect repository. Process model to represent Different phases, Lifecycle models

Unit – II: Levels of testing **(07)**

Need for levels of testing, Unit testing, Integration testing, System Testing - Usability and Accessibility Testing, Configuration Testing, Compatibility Testing, Stress testing, Regression testing, Alpha, Beta and Acceptance testing.

Unit – III: Testing techniques **(07)**

Using White Box Approach to Test design - Static Testing, Structural Testing, Unit Functional Testing, Challenges in White box testing, Using Black Box Approaches to Test Case Design, Random Testing, Requirements based testing, Decision tables, State-based testing, Cause-effect graphing, Error guessing, Compatibility testing.

Unit – IV: Fundamentals of software quality assurance **(07)**

SQA basics, Components of the Software Quality Assurance System, software quality in

business context, planning for software quality assurance, product quality and process quality, software process models, 7 quality control Tools and Modern Tools.

Unit – V: Quality assurance models (07)

Models for Quality Assurance, ISO-9000 series, CMM, CMMI, Test Maturity Models, SPICE, Malcolm Baldrige Model- P-CMM, Clean-room software engineering ,Defect Injection and prevention, Inspections & Walkthroughs, Case Tools and their effect on Software Quality.

Unit – VI: Software test automation and current industry trends (07)

Software Test Automation, Skills needed for Automation, Scope of Automation, Design and Architecture for Automation, Requirements for a Test Tool, Challenges in Automation Tracking the Bug. Combining Manual and Automated Testing, Adoption of DevOps, Big Data Testing, IoT Testing, Introduction to testing tools.

Text Books:

1. Srinivasan Desikan, Gopalaswamy Ramesh,Software Testing: Principles and Practices,Pearson
2. Ilene Burnstein, Practical Software Testing, Springer International edition

Reference Books:

1. Paul C. Jorgensen, Software Testing: A Craftsman's Approach, Auerbach Publications
2. William Perry, Effective Methods of Software Testing, Wiley Publishing, Third Edition
3. Stephen Kan, Metrics and Models in Software Quality, Addison – Wesley, Second Edition
4. Watts S Humphrey, Managing the Software Process, Pearson Education Inc.

IT 4103 Software Architecture and Design Pattern Laboratory

Teaching Scheme:
Practical: **2** hours/week
Tutorial: NIL

Examination Scheme:
Oral: **50** marks
Credits: **1**

Prerequisites: Web Engineering Technology, Programming skill development laboratory

Course Objectives:

Familiarize students with

1. One client side programming Technology
2. One server side programming Technology
3. Developing a multiuser application

Course Outcomes:

Students should be able to

1. Apply appropriate technology to design the client side of the application
2. Apply appropriate technology to design the server side part of the application
3. Design the persistent layer classes their connection to database
4. Deploy and run the complete application

List of Assignments

1. Identify a system having three or four user expectations. Prepare its use case model
2. For the same system, prepare its analysis class model.
   Implement it using java language
3. For the same system, refine the analysis model and prepare the design class model.
   Implement it using java. Include appropriate applicable design patterns while designing the system.
4. Add view classes to your model and run the code handling appropriate events.
5. Design persistent layer classes and connect the business logic to database.
6. Deploy the application on server and ensure that it runs for various clients.
   Comment on the Quality attributes addressed in the system.

Text Books

1. Robert Sebasta, Programming the world wide web, Pearson Education, Edition 7, 2013
2. Deitel, Deitel and Nieto, Internet and World wide web how to program
   Pearson Education, Edition 5, 2013

Reference Books

1. Kogent Learning Solutions Inc Web Technologies Black Book 2009

IT 4104-PROJECT PHASE – I

Teaching Scheme:
Tutorial: **2** hrs/week
Practical:**14** hrs/week

Examination Scheme:
In semester: **100** marks
Oral: **50** marks
Credits: **9**

Course Objectives :
Familiarize students with:

1. The practical implementation of theoretical knowledge gained till date.
2. implementation of their ideas/real time industrial problem/ current application of Computer Science or Information Technology.

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

1. Formulate a statement for the problem in Computer Science or Information Technology domain.
2. Prepare prototype for the identified problem.
3. Prepare System Specifications.
4. Work in team using ethical practices.

Following activities are expected to be completed in Project Phase-I:

1. Identification of Problem
2. Feasibility study
3. Formulation of Problem Statement
4. Abstract writing
5. Literature Survey
6. Project planning and maintaining log
7. High level System Design
8. Preparation of UML diagram using Tools.
9. Study of technology/platform
10. Technical Report writing
11. Start project based online course.

All this should be done with frequent meetings with internal and external guide.
The log has to be maintained.
Every project group has to give 2 Reviews in Semester-I
In Review-I, Point 1 to 4 should be completed. Demonstration and discussion with reviewers will be done.
In Review-II, Point 5 to 11 should be completed. Demonstration and discussion with reviewers will be done.

Final Year B. Tech. (IT) Semester – II										
Course Code	Course Title	Teaching Scheme Hours /Week			Examination Scheme				Marks	Credit
		Lecture	Tutorial	Practical	In Semester	End Semester	Oral	Practical		
IT 4201	Information and Cyber Security	3	0	0	50	50	0	0	100	3
PEIT 4201	Program Elective-I	3	0	0	50	50	0	0	100	3
OE 4201	Open Elective-II	3	0	0	50	50	0	0	100	3
IT 4202	Information and Cyber Security Laboratory	0	0	2	0	0	50	0	50	1
IT 4203	Project Phase-II	0	2	16	100	0	50	0	150	10
IT 4204	Project based Online Course**	2	0	0	50	0	0	0	50	2
	Total	**11**	**2**	**18**	**300**	**150**	**100**	**0**	**550**	**22**
	Grand Total	**31**			**550**				**550**	**22**

****The student shall register and complete the project based online course preferably in semester- I but may complete the same till the end of semester-II.**

Program Elective I

1. Principles of Compiler Design
2. Information Retrieval
3. Internet of Things
4. Software Defined Networks

College of Engineering for Women Karvenagar, Pune-411052 MKSSS

DEAN ACADEMICS
MKSSS's Cummins College of Engineering for Women Karvenagar, Pune-411052

Principal
MKSSS's Cummins College of Engg. For Women, Karvenagar, Pune-52.

APPROVED BY
Governing Body Members
MKSSS's Cummins College of Engineering for Women Karvenagar, Pune-411052

IT 4201 Information and Cyber Security

Teaching Scheme:
Lectures: **3** hours/week
Tutorial: NIL

Examination Scheme:
In-Semester: **50** marks
End-Semester: **50** marks
Credits: **3**

Prerequisites: Foundations of Computer Networks, Computer Networks

Course Objectives:

Familiarize students with

1. Information Security course surveys central concepts in applied information security and cyber security.
2. Make students aware of the major security risks and attack vectors.
3. Provides tools and practices for building secure systems.
4. Design, develop and support a global security system using the state of mind and reasoning on software systems security.

Course Outcomes:

Students should be able to

1. Implement the cipher techniques.
2. Analyze the various security algorithms and protocols.
3. Use different open source tools for network security and analysis.
4. Develop security systems.

Unit – I Cryptographic Techniques and Algorithms I (07)

Classical Encryption Techniques, Block Ciphers and DES, Basic Concepts in Number Theory and Finite Fields, Advanced Encryption Standard (AES), Block Ciphers. Operations,

Unit – II Cryptographic Techniques and Algorithms II (07)

Pseudo Random Number Generation and Stream Ciphers , Public Key Cryptography, Cryptographic Hash Functions Message Authentication Codes

Unit – III Cryptographic Protocols I (07)

Digital Signatures, Public-Key Certificates PKI, PKIX, and X.509, CA Hierarchy , User Authentication Protocols Public-Key Certificates PKI, PKIX, and X.509, CA Hierarchy

Unit – IV Cryptographic Protocols II (07)

Advanced Protocols: Zero knowledge Proofs, Identity based public key, Secure elections, Secure multi-party computation, and Digital cash.

Unit – V **Network Security** **(07)**

IP Security , Transport Level Security (TLS) HTTPS, HTTPS Use, Secure Shell (SSH), SSH Protocol Stack, Wireless Network Security, Wireless Network Threats, Countermeasures Mobile Device Security

Unit – VI **Cyber security** **(07)**

Electronic Mail Security: Email Security Enhancements, Pretty Good Privacy (PGP), S/MIME Intrusion Detection Malicious Software

Text Books

1. William Stallings, "Cryptography and Network Security: Principles and Practice," 6th Edition, Pearson.

Reference books

1. D. R. Stinson: Cryptography: Theory and Practice (Discrete Mathematics and Its Applications), 3e, CRC Press.
2. B. Schneier: Applied cryptography: protocols, algorithms, and source code in C, 2e, John Wiley & Sons.
3. Bernard Menezes: Network Security & Cryptography, 1st Edition, Cengage Learning, Delhi, 2011.

PEIT 4201 Principles of Compiler Design

Teaching Scheme:
Lectures: **3** hrs / week
Tutorial: NIL

Examination Scheme:
In-Semester: **50** marks
End-Semester: **50** marks
Credits: **3**

Prerequisites: Data Structures, Theory of Computation, Operating System

Course Objectives:
Familiarize students with

1. Process of compilation.
2. Tools used for the development of compilers and other language translation softwares.
3. Basic issues in code generation and optimization.

Course Outcomes:
Students should be able to

1. Design a lexical analyzer for a subset of C language.
2. Design a syntax analyzer for a subset of C language.
3. Generate intermediate code for the given programming language construct.
4. Apply different code optimization & generation techniques for a given code.

Unit – I: Introduction to Compiler & Lexical Analysis (07)

Introduction to compilers Design issues, passes, phases, symbol table Preliminaries, Memory management, Lexical Analysis Tokens, Regular Expressions, Process of Lexical analysis, Block Schematic, Automatic construction of lexical analyzer using LEX, LEX features and specification

Unit – II: Syntax Analysis (07)

Syntax Analysis Grammar (ambiguous, unambiguous, CFG), top-down parser (RDP, Predictive) and bottom-up parsers (SLR, LR-1, LALR), Error detection and recovery, automatic construction of parsers using YACC

Unit – III: Semantic Analysis (07)

Introduction to Semantic analysis, Need of semantic analysis, type checking, Syntax directed translation scheme, Intermediate code - need, types: Syntax Trees, DAG, Three Address codes: Quadruples, Triples and Indirect Triples, Intermediate code generation of declaration statement and assignment statement.

Unit – IV: Runtime Storage Management **(07)**

What is run-time support? Parameter passing methods, Storage allocation, Activation records, Static scope and dynamic scope, Heap memory management, Garbage Collection

Unit – V: Code optimization **(07)**

Machine Independent: Peephole optimizations: Common Sub-expression elimination, Removing of loop invariants, Induction variables and Reduction in strengths, use of machine idioms

Unit – VI: Code Generation **(07)**

Basic block, Register allocation and Assignment, Simple code generator, Sethi Ulman algorithm for code generation

Text Books:

1. Alfred V. Aho, Ravi Sethi, Jeffrey D. Ullman, "Compilers Principles, Techniques and Tools Addison Wesley, ISBN:981–235–885-4.
2. Kenneth C. Louden , "Compiler Construction: Principles and Practice", Course Technology In ISBN-10: 0534939724, ISBN-13: 978-0534939724.

Reference Books:

1. Dick Grune, Bal, Jacobs, Langendoen, "Modern Compiler Design", Wiley, ISBN 81-265-041 8.
2. J R Levin, T Mason, D Brown, "Lex and Yacc", O'Reilly, 2000 ISBN 81-7366-061-X.
3. K Muneeswaran, "Compiler Design", Oxford University press, ISBN 0-19-806664-3.
4. Allan Holub, "Compiler design in C", Prentice Hall, ISBN-13: 978-0131550452, ISBN-1 0131550454.

PEIT 4201 Information Retrieval

Teaching Scheme:
Lectures: **3** hrs/week
Tutorial: NIL

Examination Scheme:
In-Semester: **50** marks
End-Semester: **50** marks
Credits: **3**

Prerequisites: Data structures

Course Objectives:

Familiarize students with

1. Concepts if Information Retrieval System.
2. Indexing techniques of Information retrieval System
3. Clustering in information retrieval system
4. Understand information sharing on semantic web

Course Outcomes:

Students should be able to

1. Model the working of information retrieval search system
2. Analyze Search Strategies used in Information retrieval system
3. Evaluate Information retrieval system using different statistical measures
4. Design techniques for information retrieval system

Unit – I: Introduction **(07)**

Basic Concepts of Information Retrieval, IR system architecture. Automatic Text Analysis: Luhn's ideas, Conflation Algorithm, Porter Stemmer, Retrieval Evaluation: Precision, Recall, F-Score, Mean Average Precision, Mean Reciprocal Rank, User oriented measures

Unit – II: Indexing and Clustering **(07)**

Indexing and Index Term Weighing, Probabilistic Indexing, Inverted file, Suffix trees & suffix arrays, Signature Files, Clustered files, Cluster Hypothesis, Clustering Algorithms: Single Pass Algorithm, Single Link Algorithm

Unit – III: Search Strategies **(07)**

Retrieval strategies: Vector Space model, Probabilistic retrieval strategies, Language models, Inference networks, Extended Boolean retrieval, Latent semantic indexing, Fuzzy set retrieval

Unit – IV: Web Mining **(07)**

Searching the Web: Challenges, Characterizing the Web, Search Engines, Browsing, Mata-searchers, Web crawlers, Meta-crawler, Web data mining, Finding needle in the Haystack, Searching using Hyperlinks

Unit – V: Semantic Search Systems **(07)**

Semantic Search systems, Semantic Web, Ontology, Searching across ontologies, semantic web search, Google knowledge graphs

Unit – VI: Trends In Information Retrieval **(07)**

Case Study: Google Analytics, Search Engine Optimization, Ranking Algorithms, Recommendation Systems: Collaborative Filtering.

Text Books:

1. Yates & Neto, Modern Information Retrieval, Pearson Education, ISBN:81-297-0274-6
2. C.J. Rijsbergen, Information Retrieval, (www.dcs.gla.ac.uk).,2ndISBN:978-408709293

Reference Books:

1. Grigoris Antoniou and Frank van Harmelen, A semantic Web Primer, Massachusetts.
2. Christopher D. Manning, Prabhakar Raghavan and Hinrich Schutzen, Introduction to Information Retrieval, Cambridge University Press, Online book, ISBN:978-0-521-86571-5.

PEIT 4201 Internet of Things

Teaching Scheme:
Lectures: **3** hours/week
Tutorial: NIL

Examination Scheme:
In-Semester: **50** marks
End-Semester: **50** marks
Credits: **3**

Prerequisites: Computer Networks

Course Objectives:

Familiarize students with

1. Logical and Physical design of IOT
2. IOT architecture and its structural aspects
3. Various IOT Protocols
4. IOT solutions and applications

Course Outcomes:

Students will be able to:

1. Interpret logical and physical design of IOT enabling technologies
2. Link IOT architecture with its different structural aspects
3. Differentiate various IOT protocols
4. Propose IOT solutions for various applications

Unit – I Introduction (07)

Definition and characteristics of IOT, Physical design of IOT: Things in IOT, IOT Protocols, Logical Design of IOT: IOT functional blocks, Logical Design of IoT: Functional block, communication Model, Communication API's, IoT Enabling Technologies

Unit – II IOT Network Architecture (07)

IOT Architecture, IoT levels and Deployment templates: Level 1 to Level 5.Introduction to M2M, Difference between IoT and M2M, IoT protocol stack, Fog Computing, Edge Computing

Unit – III IOT Physical Devices and Objects (07)

Basic building blocks of IOT Device, Sensors, Actuators, and Smart Objects, Exemplary Devices: Raspberry Pi, Raspberry Pi Interfaces, pcDuino, Beagle Bone Black, CubieBoard, ARDUINO, SCADA

Unit – IV IOT Networking and Addressing techniques (07)

RFID technology, Wireless Sensor Networks, IPv6 Protocol Overview, comparison of IPv4 and IPv6, IPV6 tunneling, IPsec in IPv6, Quality of Service in IPv6

Unit – V IOT Protocols and Cloud offerings (07)

IoT Access Technologies: IEEE 802.15.4, IEEE 802.15.4g and 802.15.4e, IEEE 1901.2a, LoRaWAN, MQTT protocol

Introduction to cloud storage models and communication API's, web services for IoT

Unit – VI IOT Applications (07)

Smart City, Agriculture, healthcare, Retails, Environment

Text Books

1. “Internet of Things: A Hands-On Approach”, Arshdeep Bahga, Vijay Madisetti, University Press, 2015, ISBN: 978- 8173719547.
2. “Building the Internet of Things with IPv6 and MIPv6: The Evolving World of M2M Communications”, Daniel Minoli, Wiley Publications, 2013, ISBN: 978-1-118- 47347-4

Reference Books

1. “The Internet of Things: Connecting Objects to the Web”, Hakima Chaouchi, Wiley Publications, ISBN: 978-1- 84821- 140-7.
2. “IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things”, David Hanes, Gonzalo Salgueiro, Patrick Grossetete, Cisco Press, 16 Aug 2017, ISBN: 978-1- 58714-456- 1 599.

PEIT 4201 Software Defined Networks

Teaching Scheme:
Lectures: **3** hours/week
Tutorial: NIL

Examination Scheme:
In-Semester: **50** marks
End-Semester: **50** marks
Credits**: 3**

Prerequisites: Foundations of Computer Networks, Computer Networks

Course Objectives:

1. Appraise SDN
2. To comprehend role of data, control, and management planes and their separation
3. Differentiate between network virtualization and network function virtualization
4. Analyze Openflow protocol.

Course Outcomes:

Students should be able to

1. To develop conceptual design of SDN solutions.
2. To apply network virtualisation for industry standard solutions
3. To solve industry case-studies based on SDN.
4. Analyse the functions and components of the SDN architecture.

Unit – I SDN architecture and Fundamentals. (07)

Introduction: The Modern Data Center, Roles and Separation of data, control and management Planes, Advantages and Disadvantages. Need of SDN, Genesis of SDN.
Working of SDN: Fundamental characteristics, SDN Devices, SDN controllers, Applications

Unit – II Openflow and Abstraction (07)

Introduction: Definition, OpenFlow architecture, Flow & Group Tables, types, Hybrid Approaches, The OpenFlow forwarding and pipeline model. OpenFlow Advantages and Limitations, OpenFlow Protocol.

Unit – III Network Virtualization (07)

Definition, Concepts, Benefits of Network Virtualization, Components of a Virtual Network, Applications, Existing Network Virtualization Framework

Unit – IV Control Plane (07)

Control Plane: Overview, Existing SDN Controllers including Floodlight and Open Daylight projects. Customization of Control Plane: Switching and Firewall Implementation using SDN Concepts.

Unit – V Data Plane (07)

Data Plane: Software-based and Hardware-based; Programmable Network, Hardware. Programming SDNs: Northbound Application Programming Interface, Current Languages and Tools, Composition of SDNs.

Unit – VI Network Function Virtualization (07)

Introduction: Concepts, Comparison of NFV and NV, Implementation and Application, Data Center Networks, Application of NFV in LTE, IMS, Content Delivery, Mobile Networks

Text Books

1. Thomas D. Nadeau, Ken Gray, SDN: Software Defined Networks, An Authoritative Review of Network Programmability Technologies, O'Reilly Media, ISBN:10:1-4493-4230-2, 978-1-4493-4230-2

2. Paul Goransson and Chuck Black, Software Defined Networks: A Comprehensive Approach, Morgan Kaufmann, ISBN:9780124166752, 9780124166844

Reference books

1. Vivek Tiwari, SDN and OpenFlow for Beginners, Digital Services, 10: 1-940686-00-8 13: 978-1-940686-00-4
2. Fei Hu, Network Innovation through OpenFlow and SDN: Principles and Design,CRC
   Press,ISBN:10: 1466572094

3. Open Networking Foundation (ONF)Documents, https://www.opennetworking.org

4. Online Reading, http://www.nec-labs.com/~lume/sdn-reading-list.html,

OE 4201 Unified Communication

Teaching Scheme:
Lectures: **3** hours/week

Examination Scheme:
In-Semester: **50** marks
End-Semester: **50** marks
Credits: **3**

Prerequisites: Foundations of Computer Networks, Computer Networks

Course Objectives:
Familiarize students with

1. Compare Circuit switching and packet switching related to performance parameters.
2. Choose VOIP protocols for unified communications.
3. Analyze contact center as application of unified communications.
4. Interpret emerging technologies/protocols in VOIP communications.

Course Outcomes:
Students should be able to

1. Apply VOIP unified communications and analytics concepts to Contact Center Working.
2. Design and Implement VOIP protocols for telecommunication systems/applications.
3. Interpret and apply current or emerging knowledge in telecommunication engineering.
4. Use relevant mathematics and computer science concepts as tools.

Unit – I Introduction to digital and lP Telephony (07)

Digital Telephony: circuit switched networks, ss7, ISDN, Exchanges, E.164 Numbering Plans IP Telephony: Packet switched Networks, signaling & Media separation' Media Encapsulation ' RTP and RTCP, Audio and Video Codecs.

Unit – II VolP Protocols (07)

H.323 Network Elements, H.323 protocol, H.323 Call flows, SIP Network Elements, SIP Protocols, SIP Call Flows, H.248 protocol : Media Gateways, Media Gateway controllers, commands, Transactions, Contexts, Terminations, Descriptors' Packages

Unit – III Unified Communications (07)

Local and Network features, Voice & Data Integration, Collaboration, Mobility, Business Applications: Framework for custom applications, computer Telephony Interface, Application Sequencing.

Unit – IV Inbound Contact Center (07)

Call Centers: Introduction, Evolution and classification of Contact Centers.
Inbound Contact Center :Introduction Self Service / Interactive Voice Response, Routing, Intelligent Routing, VXML
Agent : Skills, Selection Algorithms, Modes, Service Observing, Recording

Unit – V Outbound Contact Center and Reporting (07)

Outbound contact center: Introduction, Proactive contact: voice, SMS, E-mail & chat. Contact Center Reporting: Types of Reports, Business use cases.
Analytics: Agent Performance, Occupancy

Unit – VI Emerging technologies in Telecommunications (07)

High Availability: Load balancing, Reliability, Failover & Failback, Location Redundancy, Hardware footprint, cloud Computing, Emerging Technologies: Google Glass, WebRTC, Hosting on Cloud.

Text Books

1. Allan Sulkin, "PBX Systems for IP Telephony" McGraw-Hil! Professional

Reference books

1. ITU-T H.323 Packet-based multimedia communications systems
2. ITU-TH.225Call Signaling Protocols and media stream packetization
3. ITU-T H-245 Control protocol for multimedia communication
4. IETF RFC 326131P: Session Initiation Protocol
5. IETF RFC4566 SDP: Session Description Protocol
6. Contact Center for' Dummies, Wiley Publishing Inc.
7. Real Time Communication with WebRTC, O'Reilly Publishing

IT 4202 Information and Cyber Security Laboratory

Teaching Scheme: **Examination Scheme:**

Practical: **2** hours/week Oral exam: **50** marks

Credits: **1**

Prerequisites: Foundations of Computer Networks, Computer Networks

Course Objectives:

1. Learn to implement the algorithms DES, RSA, MD5, SHA-1 etc.
2. Make students aware of the major security risks and attack vectors.
3. Provides tools and practices for building secure systems.
4. Learn to use network security like GnuPG, KF sensor, Net Strumbler

Course Outcomes:

Students should be able to

1. Implement the cipher techniques
2. Analyze the various security algorithms and protocols
3. Use different open source tools for network security and analysis
4. Develop security systems.

List of experiments:

1.Implement the following SUBSTITUTION & TRANSPOSITION TECHNIQUES concepts (any 2) :

a) Caesar Cipher
b) Playfair Cipher
c) Hill Cipher
d) Vigenere Cipher
e) Rail fence – row & Column Transformation

2. Implement the following algorithms (any 3)

a) DES
b) RSA Algorithm
c) Diffie-Hellman
d) MD5
e) SHA-1

3.Implement the Signature Scheme

4. Demonstrate intrusion detection system (ids) using any tool (snort or any other s/w)

5. Analysis the Security Vulnerabilities of E-commerce services. / Analysis the security vulnerabilities of E-Mail Application

6. Steps to ensure Security of any one web browser (Mozilla Firefox/Google Chrome)

7. Study assignment: (any 1)
   A. Study of different wireless network components and features of any one of the Mobile Security Apps.
   B. Study of the features of firewall in providing network security and to set Firewall Security in windows.
   C. Study of different types of vulnerabilities for hacking a websites / Web Applications.

Text Books

1. William Stallings, "Cryptography and Network Security: Principles and Practice," 6th Edition, Pearson.

Reference books

1. B. Schneier: Applied cryptography: protocols, algorithms, and source code in C, 2e, John Wiley & Sons.
2. Bernard Menezes: Network Security & Cryptography, 1st Edition, Cengage Learning, Delhi, 2011.

IT 4203-Project Phase – II

Teaching Scheme:
Tutorial: **2** hrs/week
Practical: **16** hrs/week

Examination Scheme:
In-Semester: **100** Marks
Oral: **50** marks
Credits: **10**

Prerequisites: BTech-Project Phase I – Semester I

Course Objectives:
Familiarize students with:

1. Product development cycle.
2. Paper presentation activities.
3. Technical writing.

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

1. Propose a System model.
2. Apply technical knowledge for solving a problem.
3. Create solution for problem in Computer Science and Information Technology.
4. Test product/service.

Following activities are expected to be completed in Project Phase-II:

1. Completion of online course.
2. Additional literature survey.
3. Detailed System design.
4. Implementation of project.
5. Write test cases.
6. Test developed project using testing tools.
7. Writing journal/conference paper on the project.
8. Participations in project competitions.
9. Project report preparation.

All this should be done with frequent meetings with internal and external guide.
The log has to be maintained.
Every project group has to give 2 Reviews in Semester-II
In Review-III, Point 1 to some part of 4 should be completed. Demonstration and discussion with reviewers will be done.
In Review-IV, remaining part from Point 4 to 9 should be completed. Demonstration and discussion with reviewers will be done.
At least one paper should be published in reputed International conference/International journal.

IT 4204 Project Based Online Course

Teaching Scheme:
Lecture: **2** hrs/week

Examination Scheme:
In-Semester: **50** marks
Credits: **2**

Course Objectives:

Familiarize students with
1. Exploring technical literature with the purpose of formulating a project statement.
2. Formulate intended future work based on the course they have registered.
3. Developing a prototype for the project statement.

Course Outcomes:

Students should be able to
1. Perform focused study of technical literature relevant to a specific topic.
2. Build independent thinking abilities to approach complex problems.
3. Extract desired knowledge from Online course.
4. Apply course knowledge for implementing the project.

Contents

1. The Project guide will suggest one/two online courses (which students have not studied till date)
2. Multiple courses can be taken by different group member of the same group.
3. Assignments related to project should be completed.