PRASAD V. POTLURI
SIDDHARTHA INSTITUTE OF TECHNOLOGY
(AUTONOMOUS)
Kanuru.Vijayawada-520007
AICTE approved, NBA & NAAC accredited, An ISO 9001-2008 certified Institution
Permanent Affiliation to JNTUK, Kakinada.
Ph: 0866-2581699, e-mail: principal@pvpsiddhartha.ac.in, web: www.pvpsiddhartha.ac.in

COMPUTER SCIENCE & ENGINEERING

SYLLABUS BOOK

(PVP 14)

B.TECH FOUR YEAR DEGREE COURSE

Sponsored by
Siddhartha Academy of General & Technical Education
VIJAYAWADA

Department Vision

To be a centre of excellence in Computer Science and Engineering and take up challenge for the benefit of society.

Department Mission

a. Impart professional education through best curriculum in harmony with the industry needs

b. Inculcate ethics, research capabilities and team work in the young minds so as to put efforts to the advancement of the nation.

c. Strive for student achievement and success with leadership qualities and preparing them for continuous learning in the global environment.

Program Educational Objectives (PEOs)

PEO-I The graduates of the program will excel in the fundamental concepts of basic engineering and advanced concepts of computer science engineering.
PEO-II The graduates of the program will be professional in computing industry or pursuing higher studies.
PEO-III The graduates of the program will excel in team work, ethics, and communication skills and contribute to the benefit to the society.

Program Outcomes (POs)

At the end of the program, the graduate will have:

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

b. An ability to analyze a problem, identify and define the computing requirements to its solution.

c. An ability to design, implement and evaluate a computer-based system, process, component or program.

d. An ability to function effectively as an individual and as a member in a team.

e. To understand professional ethics and social concerns.

f. A knowledge of contemporary issues for developing and managing projects economically.

g. An ability to communicate effectively in both verbal and written forms.

h. An ability for life-long learning in the context of technological changes.

i. An ability to use the techniques, skills and modern engineering tools necessary for engineering practice.

j. An ability to design, develop principles and construction of software/hardware systems of varying complexities.

k. A Knowledge to solve problems on economic, environmental concerns in societal context.

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Course Structure (Effective from Academic Year 2014-15)

1/4 B.Tech – First Semester

Course Code	Subject	Contact Hours		Marks			Credits
		THEORY	LAB	INT	EXT	TOTAL	
CS1T1	Engineering Mathematics – I	3 + 1 *		30	70	100	3
CS1T2	English for Communication	3		30	70	100	3
CS1T3	Engineering Chemistry	3 + 1 *		30	70	100	3
CS1T4	Environmental Studies	3		30	70	100	3
CS1T5	Basic Electrical Engineering	3 + 1 *		30	70	100	3
CS1T6	Introduction to Computers	3 + 1 *		30	70	100	3
CS1L1	English Language communication Skills Lab		3	25	50	75	2
CS1L2	IT Workshop		3	25	50	75	2
CS1L3	Engineering Graphics Lab		3	25	50	75	2
	Total	22	9	255	570	825	24

* Indicates Tutorial

1/4 B.Tech – Second Semester

Course Code	Subject	Contact Hours		Marks			Credits
		THEORY	LAB	INT	EXT	TOTAL	
CS2T1	Engineering Mathematics – II	3 + 1 *		30	70	100	3
CS2T2	Professional Ethics	3		30	70	100	3
CS2T3	Engineering Physics	3 + 1 *		30	70	100	3
CS2T4	Digital Logic Design	3 + 1 *		30	70	100	3
CS2T5	Basic Electronics Engineering	3 + 1 *		30	70	100	3
CS2T6	C Programming	3 + 1 *		30	70	100	3
CS2L1	Engineering Physics & Chemistry Lab		3	25	50	75	2
CS2L2	Basic Electronics Engineering Lab		3	25	50	75	2
CS2L3	C Programming Lab		3	25	50	75	2
	Total	23	9	255	570	825	24

* Indicates Tutorial

2/4 B.Tech – First Semester

Course Code	Subject	Contact Hours		Marks			Credits
		THEORY	LAB	INT	EXT	TOTAL	
CS3T1	Discrete Mathematics	3 + 1 *		30	70	100	3
CS3T2	Data Structures	3 + 1 *					
CS3T3	Program Design	3 + 1 *		30	70	100	3
CS3T4	Formal Languages and Automata Theory	3 + 1 *		30	70	100	3
CS3T5	Object Oriented Programming through Java	3 + 1 *		30	70	100	3
CS3L1	Data Structures Lab		3	25	50	75	2
CS3L2	Advanced C Programming Lab		3	25	50	75	2
CS3L3	JAVA Lab		3	25	50	75	2
CS3L4	Technical English		2	50		50	1
	Total	20	11	275	500	775	22

* Indicates Tutorial

2/4 B.Tech – Second Semester

Course Code	Subject	Contact Hours		Marks			Credits
		THEORY	LAB	INT	EXT	TOTAL	
CS4T1	Compiler Design	3 + 1 *		30	70	100	3
CS4T2	Design and Analysis of Algorithms	3 + 1 *		30	70	100	3
CS4T3	File Structures	3 + 1 *		30	70	100	3
CS4T4	Principles of Programming Languages	3 + 1 *		30	70	100	3
CS4T5	Computer Organization	3 + 1 *		30	70	100	3
CS4L1	Compiler Design Lab		3	25	50	75	2
CS4L2	File Structures Lab		3	25	50	75	2
CS4L3	Computer Organization Lab		3	25	50	75	2
CS4L4	Personality Development Course		2	50		50	1
	Total	20	11	275	500	775	22

* Indicates Tutorial

3/4 B.Tech – First Semester

Course Code	Subject	Contact Hours		Marks			Credits
		THEORY	LAB	INT	EXT	TOTAL	
CS5T1	Database Management Systems	3 + 1 *		30	70	100	3
CS5T2	Microprocessor and Interfacing	3 + 1 *		30	70	100	3
CS5T3	Computer Networks	3 + 1 *		30	70	100	3
CS5T4	Soft Computing	3 + 1 *		30	70	100	3
CS5T5	Operating Systems	3 + 1 *		30	70	100	3
CS5L1	Database Management Systems Lab		3	25	50	75	2
CS5L2	Microprocessors Lab		3	25	50	75	2
CS5L3	Computer Networks & Operating Systems Lab		3	25	50	75	2
CS5L4	Free Open Source Software Tools		3	25	50	75	2
	Total	20	12	250	550	800	23

* Indicates Tutorial

3/4 B.Tech – Second Semester

Course Code	Subject	Contact Hours		Marks			Credits
		THEORY	LAB	INT	EXT	TOTAL	
CS6T1	Advanced Java & Web Technologies	3 + 1 *		30	70	100	3
CS6T2	Design Patterns	3 + 1 *		30	70	100	3
CS6T3	Computer Graphics	3 + 1 *		30	70	100	3
CS6T4	Data Warehousing & Data Mining	3 + 1 *		30	70	100	3
CS6T5	Free Elective	3 + 1 *		30	70	100	3
CS6L1	Advanced Java and Web Technologies Lab		3	25	50	75	2
CS6L2	UML & Design Patterns Lab		3	25	50	75	2
CS6L3	Computer Graphics Lab		3	25	50	75	2
CS6L4	Soft Skills Course		2	50		50	1
CS6L5	Seminar		2	50		50	1
	Total	20	13	325	500	825	23

* Indicates Tutorial

Free Elective

CS6T5FE1. Database Management Systems
CS6T5FE2. Web Technologies
CS6T5FE3. Object Oriented Programming through Java
CS6T5FE4. Operating Systems

4/4 B.Tech – First Semester

Course Code	Subject	Contact Hours		Marks			Credits
		THEORY	LAB	INT	EXT	TOTAL	
CS7T1	Big Data Concepts	3 + 1 *		30	70	100	3
CS7T2	Mobile Application Development	3 + 1 *		30	70	100	3
CS7T3	Information Security	3 + 1 *		30	70	100	3
CS7T4	Elective –I	3 + 1 *		30	70	100	3
CS7T5	Elective – II	3 + 1 *		30	70	100	3
CS7L1	Data Analytics Lab		3	25	50	75	2
CS7L2	Mobile Application Development Lab		3	25	50	75	2
CS7L3	Information Security Lab		3	25	50	75	2
CS7L4	Mini Project		3	75		75	2
CS7L5	Seminar		2	50		50	1
	Total	20	14	350	500	850	24

* Indicates Tutorial

Elective – I

CS7T4A. Cloud Computing
CS7T4B. Advanced Databases
CS7T4C. Image Processing
CS7T4D. Advanced Computer Architecture

Elective - II

CS7T5A. Software Engineering
CS7T5B. Embedded Systems
CS7T5C. Distributed Systems
CS7T5D. Web Services

4/4 B.Tech – Second Semester

Course Code	Subject	Contact Hours		Marks			Credits
		THEORY	LAB	INT	EXT	TOTAL	
CS8T1	Managerial Economics & Financial Analysis	3 + 1 *		30	70	100	3
CS8T2	Elective –III	3 + 1 *		30	70	100	3
CS8T3	Elective – IV	3 + 1 *		30	70	100	3
CS8PW	Major Project		9	100	200	300	9
	Total	12	9	190	410	600	18

* Indicates Tutorial

Elective - III

CS8T2A. E- Commerce
CS8T2B. Parallel Computing
CS8T2C. Software Testing Methodologies
CS8T2D. Scripting Languages

Elective - IV

CS8T3A. Human Computer Interaction
CS8T3B. TCP/IP
CS8T3C. Virtual Reality
CS8T3D. Secure Database Application Development

ENGINEERING MATHEMATICS-I (Required)
(Common to all branches during I B.Tech., I Semester)
Course Code(s): CE1T1, ME1T1, CS1T1, IT1T1, EE1T1, EC1T1, AE1T1

Credits: 3

Lecture: 3 periods/week **Internal assessment: 30 marks**
Tutorial: 1 period /week **Semester end examination: 70 marks**

Prerequisite: NIL

This course introduces the fundamental concepts of Mathematics. With this foundation students can take up engineering career in industry or research.

COURSE OBJECTIVES:

1. The main purpose of this course is to provide students with skills in solving differential equitation's, evaluating improper integral using beta and gamma functions.
2. To prepare students for lifelong learning and successful careers using mathematical concepts of differential integral and vector calculus.

COURSE OUTCOMES:
At the end of this course student will:

CO1) Acquire the knowledge of solving ordinary differential equations.

CO2) Get the knowledge of mean value theorems and able to find maxima, minima of functions of two variables.

CO3) Ability to apply double integrals to find area of the given region, triple integrals to find volume of the three dimensional objects.

CO4) Get the knowledge of finding gradient of scalar point functions, curl, divergence of vector point functions.

CO5) Get the knowledge of solving improper integrals using beta, gamma functions, able to find the curve of best fit for the given data by method of least squares**.**

UNIT-I:

ORDINARY DIFFERENTIAL EQUATIONS.

Exact equations, orthogonal trajectories, applications to Newtons Law of cooling, Law of Natural growth and decay.Non-Homogeneous linear Differential equations of second and higher order with constant coefficients with RHS term of the type , sinax, cosax, polynomials in X, V(x), x V(X).

UNIT-II:

DIFFERENTIAL CALCULUS.

Rolle's theorem, Lagrange's mean value theorem and Taylor's theorem (without proofs),Taylor's and Macluarin's series for functions of one variable. Maxima and Minima of functions of two variables, Lagrange's method of multipliers.

UNIT III:

MULTIPLE INTEGRALS.

Multiple integrals -double and triple integrals-change of variables-Change of order of Integration.

UNIT IV:

VECTOR DIFFERENTIATION &INTEGRATION.

Gradient-Divergence-Curl and their related properties of sums-products- Laplacian and second order operators(proofs of identities not included)Vector Integration -Line integral–work done–Potential function–area-surface and volume integrals Vector integral theorems: Greens, Stokes and Gauss Divergence Theorems (Without proof) and related problems

UNIT V:

CURVE FITTING &BETA GAMMA FUNCTIONS.

Fitting a straight line-Second degree curve- Exponential curve- power curve by method of least squares.

Gamma and Beta functions- properties- Evaluation of improper integrals(applications not included).

Text Books:

Higher Engineering mathematics by B.S. Grewal , khanna publishers

References:

1. Higher Engineering Mathematics,H.K.Das,S.Chand Publications.

2. Engineering Mathematics, B. V. Ramana , Tata Mc Graw Hill

e-learning resources:

http://nptel.ac.in/courses.php

http://jntuk-coeerd.in/

ENGLISH FOR COMMUNICATION(Required)
(Common to all branches during I B.Tech., I Semester)
Course Code(s): ME1T2, CE1T2, CS1T2, IT1T2, AE1T2, EE1T2, EC1T2

Credits: 3

Lecture: 3 periods/week **Internal assessment: 30 marks**
Tutorial: - **Semester end examination: 70 marks**

Prerequisite: NIL

Course Objectives:-

1. To expose the students to various socio-cultural contexts
2. To impart human values.
3. To strengthen the writing skills.
4. To enhance their communicative competence.
5. To improve their vocabulary
6. To make them well versed in grammar.
7. To enhance their comprehensive ability.

Course outcomes:-

CO1) Improved comprehensive ability
CO2) Writing skills
CO3) Acquiring human values.
CO4) Knowledge of grammar.
CO5) Cultural adaptability.

Syllabus

Unit –I

1. Unity of minds-Abdul kalam.
2. Communication
   Process of communication
   Types of communication-----Verbal and nonverbal communication.
   Listening skills.
3. Synonyms, antonyms from the prescribed syllabus.

Unit-II

1.'Next Sunday'--------R.K.Narayan
2. Tenses
3. Active/passive voice

Unit III

1.'The cop and the anthem'------O.Henry
2. Direct/Indirect speech
3. Letter writing.

Unit IV

1.'Three Questions'----Leo Tolstoy
2. Degrees of comparison
3. Reading comprehension

Unit V

1. Kalpana chawla-----Biographical sketch
2. Correction of sentences.

Reference books:

1. Communication skills ------Sanjay kumar&pushpa latha oxford.
2. Communication skills------Leena sen.(PHI)
3. English for engineering students-----G.V.L.N.Sharma.
4. An approach to communication skills----Bhanu ranjan, Dhanpat rai&co.
5. The craft of Business letter writing-----Mathew ,Tata Mac Graw Hill.

E-learning resources:

http://nptel.ac.in/courses.php

http://jntuk-coeerd.in/

ENGINEERING CHEMISTRY (Required)
(Common to CSE, IT, CE, ECE during I B.Tech, I Semester)
(Common to EEE, AE, ME during I B.Tech, II Semester)
Course Code(s): CE1T3, CS1T3, IT1T3, EC1T4, AE2T3, EE2T3, ME2T3

Credits: 3

Lecture: 3 periods/week **Internal assessment: 30 marks**
Tutorial: 1 period /week **Semester end examination: 70 marks**

Prerequisites: NIL

Course Objectives:

1. To acquire knowledge about desalination of brackish water and treatment of municipal water.
2. To gain the knowledge of conducting polymers, bio-degradable polymers and fiber reinforced plastics.
3. To learn significance of green chemistry and green synthesis and the synthesis of nano materials.

Course Outcomes:

After studying this course, students will be able to

CO1)Develop innovative methods to produce soft water for industrial use and potable water at cheaper cost.

CO2)Replace metals with conducting polymers and also produce cheaper biodegradable polymers to reduce environmental pollution.

CO3)Produce economical green synthesis and new methods of synthesis of nano materials.

CO4)Find appropriate metals or combination of metals and develop economical methods for minimizing corrosion.

CO5)Bring the new ideas in converting solar energy into most needy electrical energy efficiently and economically to reduce the environmental pollution.

UNIT – I

A)WATER TECHNOLOGY:-

Introduction, Hardness of water, types of hardness(permanent and temporary)-
Degree of hardness-Numericals-determination of hardness by EDTA Method-softening methods (line-soda, ion exchange and zeolite process)

B)WATER TREATMENT:-

Desalination-reverse osmosis-electrodialysis. Municipal water treatment-removal of micro organisms- by irradiation of UV radiation- bleaching powder process-chlorination-br`eak point of chlorination-By using chloramine-By using ozone.

UNIT - II

A)POLYMERS

Introduction - Types of polymers (addition and condensation)- mechanism of addition polymerization (free radical, ionic) – Classification - Methods of polymerisation – Stereospecific polymers - Ziegler Natta catalysis - Properties of polymers – Conducting Polymers-Engineering applications – Biodegradable polymers - Individual polymers(Preparation,Properties, Uses of Poly Styrene, PVC, PTFE, Bakelite's, Cellulose derivatives, PolyCarbonates).

B)PLASTICS :–

Types –Compounding of plastics- Moulding(Injection, compression, blow film extrusion and extrusion moulding)- Fiber reinforced plastics (Glass and carbon) – Bullet Proof Plastics– Properties of plastics – Engineering applications.

UNIT - III

A)**GREEN CHEMISTRY**:-

Introduction – Principle of green chemistry, methods of green synthesis (aqueous phase,super critical fluid extraction method, phase transfer catalyst, micro wave induced method,ultra sound method.

B)**NANO MATERIALS**:-

Introduction to Nanomaterials-preparation of few Nano materials(Carbon Nano Tubes,Fullerenes etc)-Properities of Nano materials- Engineering applications.

UNIT – IV

A)**CORROSION** -

Defination, causes and consequences of corrosion-mechanism of dry and wet corrosion-galvanicseries,Factors influencing rate of corrosion passivity of metal, types of corrosion (galvonic,differential Aeration,pitting,crevice and stress corrosion). B)**CORROSION CONTROL**:-

Cathodic protection(sacrificial anodic protection and Impressed current cathodic protection) and Application of protective coating-metallic coatings (galvanization and tinning) organic coatings (paints (mechanism not required),varnishes,lacquers and enamels).

UNIT - V

A)**SEMICONDUCTORS & SUPERCONDUCTIVTY**

SEMICONDUCTORS-Definition –Types of semiconductors (Stiochiometric,Non Stiochiometric ,Organic, Controlled Valency Semiconductors,Doping)-applications SUPERCONDUCTIVTY– Definition-Preparation –Properties –Engineering Applications.

B)**LIQUID CRYSTALS & SOLAR ENERGY**:-

LIQUID CRYSTALS-Definition –Types - applications in LCD and Engineering Applications.

SOLAR ENERGY:

Introduction – harnessing solar energy – solar heaters – photo voltaic cells – solar reflection – green house concepts.

Learning Resources

Text Books

1. A text book of Engineeringchemistry – byN.KrishnaMurthy N.Y.S.Murthy Dr.V.Anuradha.
2. A text book of Engineering chemistry –II by D.Srinivasulu, Srivastava, Roliverma.
3. A text book of Engineering chemistry by JAIN & JAIN.
4. A text book of Engineering chemistry by C.P.Murthy, C.V.Agarwal. Andra Naidu.

Reference Books

1. A text book of Engineering chemistry by S.S.DARA.
2. A text book of Engineering chemistry by Dr.C.Daniel Yesudian

e-learning resources:

http://nptel.ac.in/courses.php

http://jntuk-coeerd.in/

ENVIRONMENTAL STUDIES (Required)
(Common to EEE, CE, ME, CSE during I B.Tech., I Semester)
(Common to IT, AE, ECE during I B.Tech., II Semester)
Course Code(s): CEIT4, MEIT4, CSIT4, EE1T4, IT2T4, AE2T6, EC2T4

Credits: 3

Lecture: 3 periods/week **Internal assessment: 30 marks**
Tutorial: - **Semester end examination: 70 marks**

Prerequisites: NIL

Course Objectives:

1. To develop an awareness, knowledge, and appreciation for the natural environment.
2. To understand different types of ecosystems exist in nature.
3. To know our biodiversity.
4. To understand different types of pollutants present in Environment.
5. To know the global environmental problems.

Course Outcomes:

The student will be able to

CO1) Develop an appreciation for the local and natural history of the area.
CO2)Hope for the better future of environment in India which is based on many positive factors like Biodiversity, successive use of renewable energy resources and other resources, increasing number of peoples movements focusing on environment.
CO3) Know how to manage the harmful pollutants.
CO4)Gain the knowledge of Environment.
CO5)Create awareness among the youth on environmental concerns important in the long term interest of the society

UNIT – I

Natural Resources:

A)Forest resources – Use and over – exploitation, deforestation, case studies – Timberextraction – Mining, dams and other effects on forest and tribal people.

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

Land resources: Land as a resource, land degradation, man induced landslides,
soil erosionand desertification.

B)Energy resources: Renewable and non-renewable resources-Natural resources

and associated problems Growing energy needs, renewable and non-renewable energy sources useof alternate energy sources. Case studies.

Mineral resources: Use and exploitation problems, environmental effects of extracting and using mineral resources, case studies.

Food resources: World food problems, changes caused by agriculture and overgrazing,\effects of modern agriculture, fertilizer-pesticide problems, water logging, salinity, casestudies. Organic Farming, Bio fertilizers and Bio-pesticides

UNIT – II

A)Ecosystems:

Definition, Scope and importance, Concept of an ecosystem. - Structure and function of anecosystem. - Producers, consumers and decomposers. - Energy flow in the ecosystem - Ecological succession. - Food chains, food webs and ecological pyramids, Flow of energy,Bio-geochemical cycles, Bio-magnification, Ecosystem values, Services and carryingcapacity.

B)Biodiversity and its conservation:

Introduction - Definition: genetic, species and ecosystem diversity. Bio-geographicalclassification of India, India as a mega-diversity nation, Hot-sports of biodiversity, Value ofbiodiversity: consumptive use, productive use, social, ethical, aesthetic, option values andecosystem service values. Threats to biodiversity: habitat loss, poaching of wildlife, manwildlifeconflicts. - Endangered and endemic species of India – Conservation of biodiversity:In-situ and Ex-situ conservation of biodiversity.

UNIT - III

A)Environmental Pollution: Definition, Cause, effects and control measures of: Air pollution, Water pollution, Soil pollution, Marine pollution, Noise pollution,
Thermal pollution, Nuclear hazards

B)Solid waste Management:. - classification and characters of solid waste, factors affecting waste generation, collection and disposal of solid waste. E- waste and management. Role of an individual in prevention of pollution. – Pollution case studies.

UNIT – IV

A)Global Environmental problems and Global efforts:

Green house effect, Green house gasses, Global warming, Climate change and their impactson human environment, ozone layer depletion. International conventions / protocols: Earthsummit,Kyoto protocol & Montreal protocol.

B)Towards Sustainable Future: From Unsustainable to Sustainable development, Population and its explosion, Urban problems related to energy,Consumerism and waste products, Roleof IT in Environment and human health. Value Education. HIV/AIDS, Environmental ethics ,Concept of green buildings and Clean Development Mechanism.

UNIT - V

A)Environmental Impact Assessment & Management plans, Environmental Law: Definition ofimpact, Classification of impacts, Impacts of different components such as: human health,resources, air, water, flora & fauna. Environment management plans (EMP): Technological solutions for pollution control, Green-belt-development, Rain water harvesting, Remote sensing and GIS methods.
Environmental law (Air, Water, Wild life, Forest Acts): Objectives of Acts, Institutional arrangements for Implementation and Regulation.

B)Field work:

Visit to a local area to document environmental assets River /forestgrassland/hill/mountain-Visit to a local polluted site Urban/Rural/industrial/ Agricultural Study of common plants,insects, birds. -Study of simple ecosystems pond, river, hill slopes, etc.

Learning Resources

Text Books:

1. Erach Bharucha, 2010 "Text Book of Environmental Studies", University GrantsCommission, Universities Press (India) Pvt.Ltd., Hyderabad
2. Text Book of Environmental Sciences and Technology by M. Anji Reddy, BS Publications.

Reference:

1. Text Book of Environmental Studies by Deeshita Dave & P. Udaya Bhaskar, CengageLearning.
2. Text Book of Environmental Science and Engineering by G.Tyler Miller Jr,2006 Cengage learning
3. Text Book of Environmental Studies from Crisis to Cure by R. RajaGopalan.
4. Environmental Studies by K.V.S.G. Murali Krishna, VGS Publishers, Vijayawada

e-learning resources:

1. http://nptel.ac.in/courses.php
2. http://jntuk-coeerd.in/

BASIC ELECTRICAL ENGINEERING (Required)
(Common to CSE, IT during I B.Tech., I Semester)

Course Code(s): CS1T5, IT1T5 **Credits: 3**
Lecture: 3 periods/week **Internal assessment: 30 marks**
Tutorial: 1 period /week **Semester end examination: 70 marks**

Prerequisite: No Prerequisite

Course Objectives:

1. To impart the basic knowledge about the Electric circuit and magnetic circuits.
2. To understand the working of various DC and AC motors.
3. To know the operation of transformer.
4. To know the various electrical measuring instruments.

Course outcomes:

At the end of this course student will:

CO1) Analyze the Electrical and magnetic circuits

CO2) Understand alternating quantities and analyzing AC circuit

CO3) Understand the working of various AC & DC Motors

CO4) Understand the operation of Transformer and know the working of electrical measuring instruments

Syllabus:

UNIT - I

Introduction to Electrical Engineering: Basic Definitions, ohm's law, Voltage and current source. Kirchhoff's laws, basic circuit components, series parallel resistance circuits, mesh analysis and nodal analysis (only on independent sources).Star-Delta/Delta-Star conversion, simple problems.

UNIT-II

Magnetic Circuits: Basic definitions ,Magnetic field due to electric current flow, force on a current carrying conductor placed in a magnetic field, Faradays laws of electromagnetic induction, analogy between electric and magnetic circuits, self inductance and mutual inductance, coefficient of coupling, coils connected in series and parallel, Types of induced EMF's, Simple problems.

UNIT-III

Alternating Quantities : Principle of ac voltages , waveforms and basic definitions, relationship between frequency, speed and number of poles, root mean square and average values of alternating currents and voltage, form factor and peak factor, phasor representation of alternating quantities, analysis of ac circuits with simple basic network elements, single phase series circuits, single phase parallel circuits, power in ac circuits.

UNIT-IV

DC Machines: Principle of operation of dc motor, Torque production in a dc motor, Types of DC motors, three point starters.

A.C Machines: Construction and Principle of operation of three phase induction motor, slip, rotor frequency and torque (simple problems).

UNIT-V

Transformers: Principles of operation, Constructional Details, Ideal Transformer and Practical Transformer, Losses, Efficiency(All the above topics are only elementary treatment and simple problems).

Basic Instruments : Introduction, classification of instruments, operating principles, essential features of measuring instruments, Moving coil permanent magnet (PMMC) instruments, Moving Iron Ammeters and Voltmeters (elementary Treatment only).

TEXT BOOKS:

- Principles of Electrical Engineering by V.K Mehta, S.Chand Publications.
- Basic Electrical Engineering - By M.S.Naidu and S. Kamakshiah – TMH.

REFERENCES:

1. Theory and Problems of Basic Electrical Engineering by D.P.Kothari & I.J. Nagrath PHI.
2. Basic Electrical Engineering –By T.K.Nagasarkar and M.S. Sukhija Oxford University Press.

e-learning resources:

http://nptel.ac.in/courses.php

http://jntuk-coeerd.in/

INTRODUCTION TO COMPUTERS(Required)
(Only for CSE during I B.Tech., I Semester)

Course Code: CS1T6 **Credits: 3**
Lecture: 3 periods/week **Internal assessment: 30 marks**
Tutorial: 1 period /week **Semester end examination: 70 marks**

Prerequisite: Nil

Course Objectives:

This course intends to cover the basic concepts of computers such as organization, architecture, input and output devices, memory as well as operating systems, computer networks. Demonstrate knowledge of the main computer applications used in business and be able to choose the appropriate application for a given task.

Course Outcomes
After completion of this course, students will be able to:

CO1) Understand the computer system, software and hardware
CO2) Know the importance of different types of software
CO3) Understand how to draw flow charts and write algorithms for simple problems
CO4) Study basic concepts of computer networks and Internet
CO5) Learn fundamental concepts of Emerging Technologies

Course Contents / Syllabus:

Unit – I
Introduction to computers : what is a computer, characteristics of computers, Generation of computers, Classification of computers, Basic computer organization, Applications of computers, Input Devices, output devices, Soft copy devices, Hard copy devices.

Unit – II
Computer Memory and Processors : Introduction, Memory Hierarchy, Processor Registers, Cache memory, primary Memory, Secondary Storage devices, Magnetic
tapes, Floppy disks, Hard disks, Optical drives, USB Flash drives, Memory Cards, Mass Storage Devices, Basic Processor Architecture. Computer Software : Introduction to computer software, Classification of computer software, System software, Application software, Firmware, Middleware, Acquiring Computer Software, Design and Implementation

of Programs.

Unit - III

Operating Systems : Introduction, Evolution of Operating Systems, Popular Operating Systems, Introduction of Algorithms and Programming Languages : Algorithm, Control Structures used in Algorithms, Some more Algorithms, Flow Charts, Pseudo Code, Programming Languages, Generation of Programming Languages, Categorization of High Level Languages,

Unit - IV

Computer Networks : Introduction of Computer Networks, Connecting Media, Data Transmission Mode, Data Multiplexing , Data Switching, data routing techniques, Network Topologies, Types of Network, Networking Devices. The Internet : Internet, Internet Services, Internet Glossary, Types of Internet Connections, Internet Security

UNIT – V

Emerging Computer Technologies : Distributed Networking, Peer to peer Computing , Grid Computing, Cloud Computing, Utility Computing, On-demand Computing, Wireless network, Bluetooth, Artificial Intelligence.

Text books:

1. Fundamentals of Computers, Reema Thareja, Oxford Higher Education, Oxford University Press
2. Introduction to computers , Peter Norton, 6th Edition, Tata McGrawHill

Reference Books:

1. Computer Fundamentals , Anita Goel, Pearon Education India ,2010
2. Computer Concepts and Applications : **http://uwf.edu/clemley/cgs1570w/notes**
3. Computers in education : http://www.mhhe.com/peternorton
4. Check out Processors : http://www.pcmag.com

e-learning resources:

http://nptel.ac.in/courses.php

http://jntuk-coeerd.in/

ENGLISH LANGUAGE COMMUNICATION SKILLS LAB(Required)
(Common to CSE, IT, EEE, CE, ECE during I B.Tech., I Semester)
(Common to ME, AE during I B.Tech., II Semester)

Course Code(s): CE1L3, CS1L1, IT1L1, EE1L1, EC1L1, ME2L2, AE2L2

Credits:2

Lab: 3 periods/week **Internal assessment: 25 marks**
Semester end examination: 50 marks

Prerequisite: English Language Communication Skills

Course Objectives:-

1. To improve the communicative ability.
2. To enhance the general conversational skills in different socio-cultural contexts.
3. To strengthen their professional skills.
4. To instill confidence and make them competent enough to express themselves fluently.
5. To expose the students to various spoken skills.

Course outcomes:-

CO1) Better pronunciation and accent
CO2)Ability to use functional English
CO3)Improved comprehensive ability
CO4)Enhanced analytical skills
CO5)Good negotiation skills.

Syllabus

Task 1: Phonetics

Introduction to sounds of English.
Phonetic transcription of simple words.
Word stress or accent.
Intonation.

Task II: Spoken skills

JAM
Public speaking
Debate

Task III :-Conversation skills

Introducing
Extending Invitations
Apologizing
Lodging complaints.

Task IV: - Describing

Describing an object
Describing a process
Describing situations

Task V: - Group Discussion

Dynamics of Group Discussion
Various strategies
Discussion on various topics

Reference books:

1. Everyday dialogues in English------Robert J.Dixon.
2. Speak well------orient black swan.

IT WORKSHOP(Required)
(Common to CE, ME, CSE, IT during I B.Tech., I Sem)
(Common to AE, ECE during I B.Tech., II Sem)

Course Code(s): CE1L2, ME1L2, CS1L2, IT1L2, AE2L3, EC2L1 Credits:2
Lecture: -- Internal assessment: 25 marks
Lab: 3Hours /week Semester end examination:50marks

Prerequisite:

Course Objectives: To provide students with hands-on experience in basic hardware, productivity tools and basic operating system installations.

Course Outcomes:

At the end of this course student will:

CO1) Identify the basic computer peripherals

CO2) Acquire sufficient knowledge on assembling and disassembling a PC

CO3) Learn the installation procedure of Windows and Linux OS

CO4) Acquire knowledge on basic networking infrastructure, internet and World Wide Web

CO5) Learn productivity tools like Word, Excel and Power point

Task 1:

Identification of the peripherals of a computer:To prepare a report containing the block diagram of the CPU along with the configuration of each peripheral and its functions. Description of various I/O Devices

Task 2:

A practice on disassembling the components of a PC and assembling them.

Task 3:

Basic DOS commands, Installation of MS windows.

Task 4:

Introduction to Linux- Installation Procedure, Basic Linux Commands.

Task 5:

Hardware Troubleshooting (Demonstration): Identification of a problem and fixing the solution (improper assembly or defective peripherals).
Software Troubleshooting (Demonstration): Identification of a problem and fixing the PC for any software issues.

Task 6:

Demonstrating Importance of Networking, Transmission Media, Networking Devices Gateway, Routers, Hub, Bridge, NIC ,Bluetooth Technology, Wireless Technology, Modem, DSL, Dialup Connection.

Task 7:

MS Word Orientation: Accessing, overview of toolbars, saving files, Using help and resources, rulers, formatting ,Drop Cap , Applying Text effects, Using Character Spacing,

OLE in Word, using templates, Borders and Colors, Inserting Header and Footer, Using Date and Time option, security features in word, converting documents while saving

Task 8:

Creating project : Abstract Features to be covered:-Formatting Styles, Inserting table, Bullets and Numbering, Changing Text Direction, Cell alignment, Footnote, Hyperlink, Symbols, Spell Check , Track Changes, Images from files and clipart, Drawing toolbar and Word Art, Formatting Images, Textboxes and Paragraphs.

Task 9:

Using spread sheet features of EXCEL including the macros, formulae, pivot tables, graphical representations **Creating a Scheduler -** Features to be covered:-Gridlines, Format Cells, Summation, auto fill, Formatting Text LOOKUP/VLOOKUP

Task 10:

Performance Analysis - Features to be covered:- Split cells, freeze panes, group and outline, Sorting, Boolean and logical operators, Conditional formatting

Task 11:

Students will be working on basic power point utilities and tools which help them create basic power point presentation. Topic covered during this week includes :-PPT Orientation, Slide Layouts, Inserting Text, Word Art, Formatting Text, Bullets and Numbering, Auto Shapes, Lines and Arrows, Hyperlinks, Inserting –Images, Clip Art, Tables and Charts in Powerpoint.

Focusing on the power and potential of Microsoft power point. Helps them learn best practices in designing and preparing power point presentation. Topic covered during this week includes: - Master Layouts (slide, template, and notes), Types of views (basic, presentation, slide slotter, notes etc), Inserting – Background, textures, Design Templates, Hidden slides, OLE in PPT.

Task 12:

Students should get connected to their Local Area Network and access the Internet. In the process they should configure the TCP/IP setting and demonstrate how to access the websites and email,Customizing web browsers using bookmarks, search toolbars and pop up blockers, Search engines and their usage.

Learning Resources:

Reference Books:

1. Computer Fundamentals, Anita Goel, Pearson

2. Information Technology Workshop,3e, G Praveen Babu, M V Narayana BS Publications.

3. Introduction to Information Technology-ITL Education Solution Limited- Pearson.

4. Fundamentals of Itnformation Technology, 2nd Edition, Alexis Leon, Mathews Leon,(Leon Vikas).

ENGINEERING GRAPHICS LAB (Required)
(Common to CSE, IT, EEE during I B.Tech., I Semester)

Course Code(s): CS1L5, E1L5,AE1L5, ECL5 **Credits: 2**
Lecture: -- **Internal assessment: 25 marks**
Lab: 3 periods/week **Semester end examination: 50 marks**

Prerequisites : Basic Mathematics.

COURSE OBJECTIVES:

This course is especially beneficial for students interested in careers related to production design, engineering, drafting, commercial art, graphic arts, and construction. The class will be a complete course in drafting techniques. Topics will include lettering, sketching, orthographic representation and isometric drawing.

COURSE OUTCOMES:

At the end of this course student will:

CO1) Represent various conics, curves and scales.

CO2) Construction of orthographic projections of Points, Lines, Planes, and Solids.

CO3) Conversion of isometric views to orthographic views and orthographic views to isometric views

SYLLABUS:

UNIT - I

Polygons-Construction of Regular Polygons using given length of a side; Ellipse-General method and Oblong Methods for Construction of ellipse; Scales-Plain,Vernier and Diagonal Scales.Introduction to Orthographic Projections; Projections of Points; Projections of Straight Lines parallel to both planes; Projections of Straight Lines-Parallel to one and inclined to other plane.

UNIT - II

Projections of Straight Lines inclined to both planes, determination of true lengths, angle of inclinations and traces.

UNIT - III

Projections of Planes; Regular Planes Perpendicular / Parallel to one Reference.
Plane and inclined to other Reference Plane; inclined to both the Reference Planes.

UNIT - IV

Projections of Solids-Prisms, Pyramids, Cylinders and Cones with the axis inclined to one Plane.

UNIT - V

Conversion of Isometric Views to Orthographic Views. Conversion of Orthographic Views to Isometric Projections and Views.

Learning Resources

TEXT BOOK:

1. Engineering Drawing by N.D. Bhat, Chariot publications

REFERENCE BOOKS:

1. Engineering Drawing by M.B. Shah and B.C. Rana, Pearson publishers
2. Engineering Drawing by Dhananjay A. Jolhe, Tata McGraw Hill Publishers
3. Engineering Graphics for Degree by K.C. John, PHI Publishers

ENGINEERING MATHEMATICS -II(Required)
(Common to all branches during I B.Tech., II Semester)
Course Code(s): CE2T1, ME2T1, CS2T1, IT2T1, AE2T1, EE2T1, EC2T1

Credits: 3

Lecture: 3 periods/week **Internal assessment: 30 marks**
Tutorial: 1 period /week **Semester end examination: 70 marks**

Prerequisites: Engineering Mathematics I

COURSE OBJECTIVES:

1. After completion of this course engineers will be able to apply the concepts of matrices, Laplace transforms, Fourier series, Fourier transforms in solving engineering problems.
2. Linear algebra in the course cover material which is essential to anyone who does mathematical computation in Engineering and sciences.

COURSE OUTCOMES:

CO1) students able to solve system of Linear equations, be familiar with properties of matrices, find the inverse, Eigen values and Eigen vectors and use them in diagonalization,

CO2)Acquire knowledge in Laplace transforms, inverse Laplace transforms and how to get a solution of differential equations by using Laplace transforms.

CO3)Get knowledge of expanding a function in terms of sine and cosine functions' in Fourier series and also to get knowledge in Fourier transforms.

CO4)Get knowledge in Z-transforms, inverse Z-transforms, solving difference equations

SYLLABUS

UNIT – I

Matrices and Linear systems of equations:

Rank-Echelon form, Normal form-definition of a vector, linear independence – Solution of Linear System of equations – Direct Methods- Gauss Elimination - Gauss Jordon and Gauss Seidal Methods.

UNIT – II

Eigen values - Eigen vectors:

Eigen values - Eigen vectors - Properties – Cayley-Hamilton Theorem - Inverse and powers of a matrix by using Cayley-Hamilton theorem- some applications of eigen value problems-Diagonalization of a matrix.

UNIT – III

Laplace transforms & Inverse Laplace transforms

Laplace transforms: Laplace transforms of standard functions –Shifting Theorems, Transforms of derivatives and integrals – Unit step function –Dirac's delta function.

Inverse Laplace transforms: Convolution theorem - Application of Laplace transforms to ordinary differential equations with given initial conditions.

UNIT – IV

Fourier Series and Fourier transforms:

Fourier series: Determination of Fourier coefficients – Fourier series – even and odd functions – Fourier series in an arbitrary interval– Half-range sine and cosine series.

Fourier transforms: Fourier integral theorem (only statement) – Fourier sine and cosine integrals - Fourier transform – sine and cosine transforms – properties – inverse transforms – Finite Fourier transforms.

UNIT – V

Z-transforms:

Introduction, properties of Z-transforms-initial value theorem-final value theorem-inverse Z-transforms-applications to difference equations.

Text Books:

1. Higher Engineering Mathematics – Khanna Publishers – B.S. Grewal – 42nd Edition.
2. Advanced Engineering Mathematics – Wiley – Erwin Kreyszig- 8th Edition.

Reference:

Engineering Mathematics Vol-II, Iyengar,T.K.V, Krishna Gandhi, et.al S.Chand Co. New Delhi. **e-**

learning resources:

http://nptel.ac.in/courses.php

http://jntuk-coeerd.in/

PROFESSIONAL ETHICS(Required)
(Common to all branches during I B.Tech., II Semester)
Course Code(s): CE2T2, ME2T2, CS2T2, IT2T2, AE2T2, EE2T2, EC2T2

Credits: 3

Lecture: 3 periods/week **Internal assessment: 30 marks**
Tutorial :- **Semester end examination: 70 marks**

Prerequisites: Nil

COURSE Objectives:-

1. To inculcate the sense of social responsibility.
2. To develop a firm ethical base
3. To make the students realize the significance of ethics in professional environment.

Course outcomes:-

CO1) Improved knowledge of ethics
CO2) High sense of responsibility
CO3) Environmental awareness
CO4) Professional outlook
CO5) Developing a broad culture.

Syllabus:-

Unit I

Profession-----Definition
Three types of ethics.
Engineering ethics
Rights and responsibilities of an engineer.

Unit II

Evolution of engineering ethics
Code of ethics
Kohlberg's theory
Gilligan's theory

Unit III

Engineering as social experimentation
Engineer's social responsibility

Unit IV

Computer ethics
Ethical hacking
Privacy

Unit V

Environmental ethics.
Livable environment
Technology assessment.

Reference books: ----

1. Ethics in engineering: Mike W.Martin Roland, Mac Grow Hill.Schinzinger
2. Engineerinethics-----M.Govindarajan, S.Natarajan&V.S.Senthil Kumar. Eastern economy Edn.PHI
3. Engineering ethics---Harris pitch and Rabbins, cengage.
4. Caroline whit back---Ethics in engineering practice and research-----Cambridge.

E-learning resources:

http://nptel.ac.in/courses.php

http://jntuk-coeerd.in/

ENGINEERING PHYSICS
(Common to EEE, AE, ME, ECE during I B.Tech, I Semester)
(Common to CSE, IT, CE during I B.Tech, II Semester)

CourseCode(s): EE1T3, AE1T3,ME1T3, EC1T3, CS2T3, IT2T3, CE2T3 Credits: 3
Lecture: 3 periods/week Internal assessment: 30 marks
Tutorial: 1 period /week Semester end examination: 70 marks

Prerequisites: Nil

COURSE OBJECTIVES:

To make student understand

1. The concepts of Quantum Physics.
2. The theoretical picture about a crystal structure.
3. How to determine the different crystal structures by using X-diffraction techniques.
4. The properties of different types of solids and to have the knowledge about the energy-band diagram in the materials.
5. The advanced topics such as lasers,fibre optics and nano- materials.

COURSE OUTCOMES:

After completion of the course student will be

CO1) Acquiring the knowledge of Quantum physics the student will have the basics about the atomic scale of the systems.

CO2) Learning crystal structure and the X-ray diffraction Techniques the student could differentiate the different types of crystals.

CO3) Getting the knowledge about the different types of solids the student will know to use the appropriate solids as per requirement.

CO4) Having the knowledge about advanced topics the student will be ready to the upcoming developments in the Engineering Physics.

UNIT-I
QUANTUM PHYSICS

Planck's black body theory of radiation - Debroglie hypothesis – Properties of matter waves –G.P. Thomson experiment– Davison and Germer experiment – Heisenberg uncertainity principle –Time independent & Time dependent Schrödinger wave equation – physical significance of wave function – Particle in one dimensional potential box.

UNIT-II
Crystal Structure & X-ray Diffraction:

Introduction – Space lattice – Basis - unit cell - Lattice parameters – Bravais lattices – Crystal systems – Structure and packing fraction of simple , bcc , fcc crystals. Directions and planes in crystals – miller indices –Distance between successive parallel planes- Diffraction of X rays – Bragg's law –Laue method- Powder method.

UNIT-III
PHYSICS OF SOLIDS-I

Classical free electron theory-Quantum free electron theory- Fermi Dirac distribution function-Bloch theorem- Kronig penny model(qualitative treatment)- Classification of materials .

Dielectric constant – electronic, ionic and orientation polarizations–internal fields in solids – Clausius Mossotti relation –causes of dielectric breakdown.

UNIT-IV
PHYSICS OF SOLIDS-II

Introduction – intrinsic semiconductor and carrier concentration- Fermi level in intrinsic semiconductor conductivity in intrinsic semiconductor– extrinsic semiconductor –carrier concentration- Fermi level in extrinsic semiconductor – Drift and diffusion current – Einstein's relations – Direct and Indirect band gap semiconductors.
Origin of magnetic moment – classification of magnetic materials – Hysteresis curve – soft and hard magnetic materials- applications.

UNIT-V
ADVANCED PHYSICS

Lasers Characteristics of lasers – spontaneous and stimulated emission of radiation – population inversion – pumping – Ruby, Helium-Neon & Semiconductor lasers- Applications of lasers.

Fiber optics Principle of optical fiber – Acceptance angle and numerical aperture – Attenuation in optical fibers – applications of optical fibers.

Introduction – Surface to volume ratio- Quantum confinement effect- properties and preparation of nanomaterial – nanotubes – SWNT- MWNT- Applications of nanomaterials.

Learning Resources
Text Books:

1.Solid state Physics by S.O.Pillai. (New Age International Publications)
2.Engineering physics by M.R.Srinivasan (New Age International Publications).

Reference Books:

1.Engineering physics by D.K.Bhattacharya and A.Bhaskaran. (Oxford Publications).
2.Engineering physics by R.K Gaur and S.L. Gupta, Dhanpat Rai Publications

e-learning resources: http://nptel.ac.in/courses.php http://jntuk-coeerd.in/

DIGITAL LOGIC DESIGN(Required)

(Only for CSE during I B.Tech., II Semester)

Course Code: CS2T4 **Credits:3**

Lecture: 3 periods/week **Internal Assessment: 30 marks**

Tutorial: 1 periods/week **Semester end examination:70 marks**

Prerequisites: Introduction to Computers

Course Objectives:

1. To study the basic philosophy underlying the various number systems, negative number representation, binary.
2. Arithmetic, binary codes and error detecting and correcting binary codes.
3. To study the theory of Boolean algebra.
4. To study representation of switching functions using Boolean algebra.
5. Expressions and their minimization techniques.
6. To study the combinational logic design of various logic and switching devices and their realization.
7. To study the sequential logic circuits design both in synchronous and Asynchronous modes.
8. Logic and switching devices, their minimization techniques and their realizations.
9. To study some of the programmable logic devices and their use in realization of switching functions.

Course Outcomes:

At the end of this course student will:

CO1) Understand various types of number systems and their conversions.

CO2) Simplify the Boolean expressions and apply the Boolean theorems through logical gates

CO3) Design and implement variety of logical devices using combinational circuits concepts.

CO4) Demonstrate and compare the construction of programmable logic devices and different types of ROM

CO5) Analyze sequential circuits like Registers and Counters using flip-flops.

Course Content/Syllabus:

UNIT : I

Digital Systems and Binary Numbers: Digital Systems – Number systems and base conversions – Representation of signed Binary Numbers – Binary codes – Logic gates.

UNIT : II

Boolean Algebra : Introduction to Boolean Algebra – Axioms and Laws of Boolean Algebra – Boolean functions – Canonical and Standard Forms.

Gate – Level Minimization : Introduction – Two, Three, Four Variable K-map's – Don't Care Conditions – NAND and NOR implementation.

UNIT : III

Combinational Logic : Introduction to combinational logic circuits – Binary adder and subtractor – Look Ahead Carry Adder - Magnitude comparator – Decoders – Encoders – Multiplexers – Demultiplexers.

UNIT : IV

Memory and Programmable Logic : Introduction to Programmable Logic Devices(PLD's) – Programmable ROM(PROM) – Programmable Logic Array(PLA) – Programmable Array Logic(PAL).

UNIT : V

Synchronous Sequential Logic : Introduction to sequential circuits – Latch – Flip Flop – SR, JK, T, D Flip Flops – Flip Flop excitation tables.

Registers and Counters : Registers – Shift registers – Ripple counters – Synchronous counters – Other counters.

Learning Resources

Text Books :

1. Digital Logic and Computer Design by M. Moris Mano, 4th Edition.
2. Digital Principles and Applications by Leach, Paul Malvino, 5th Edition.

References :

1. Fundamentals of Digital Logic Design by Charles H.Roth, Jr. 5th Edition, Cengage
2. Digital Electronics by G.K. Kharate, Oxford University Press
3. Switching Theory and Logic Design by A. Anand Kumar, PHI, 2nd Edition

E-learning resources:

http://nptel.ac.in/courses.php

http://jntuk-coeerd.in/

BASIC ELECTRONICS ENGINEERING(Required)
(Common to CSE, IT during I B.Tech., II Semester)

Course Code(s): CS2T5, IT2T5 **Credits: 3**
Lecture: 3 periods/week **Internal assessment: 30 marks**
Tutorial: 1 period /week **Semester end examination: 70 marks**

Prerequisites: Nil

Course Objectives:

- To study in detail about construction of several electronic devices.
- To analyse the characteristics of various electronic devices and circuits.
- To understand the internal structure and characteristics of Op-amp.
- To learn about the linear and non-linear applications of Op-amp.

Course Outcomes:

The student will be able to

CO1) Understand the semiconductor physics of the intrinsic, p and n materials.
CO2) Understand the function and operation of diodes,rectifiers, transistors and amplifiers.
CO3) Students will be aware of the architecture, functions & their applications of IC 741 OP-Amp

UNIT – I

Semi Conductors and Diodes:

Conductors, Semiconductors, Intrinsic Semiconductors, Extrinsic Semi Conductors. Diode Theory: Basic Ideas, The ideal Diode, Forward and Reverse Bias, Diode Equation, Volt-Ampere Characteristic.

Special diodes: symbol of zener diode, operation, V-I characteristics, symbol of photo diode, working principle, LED symbol and principle.

UNIT – II

Rectifiers:

Half-wave Rectifier, Full-wave and Bridge Rectifier, derivation of Ripple factor, efficiency of Half-wave, full-wave and Bridge rectifiers. Merits and demerits of Half-wave, full-wave and Bridge rectifiers, Comparisons of rectifiers.

UNIT- III

Bipolar Junction Transistors:

Symbols of pnp and npn transistors and their working principles, Transistor currents, input and output characteristics of Common base configuration, Common Emitter configuration
Transistor Switch,

Amplifiers: working principles of Common base amplifier, Common Emitter amplifier, Common collector amplifier and their applications.

UNIT- IV

Characteristics of Op-Amps: Introduction to OP-amp, Op-amp Block Diagram, ideal and practical Op-amp specifications, 741 op-amp & its features, Op-Amp parameters & Measurement, Input & Out put off set voltages & currents, slew rates, CMRR, PSRR.

UNIT-V

Applications of Op-Amps: Inverting and Non-inverting amplifier, Integrator and differentiator, Comparators.

Learning Resources

Text Book:

1. Electronic Principles, Albert Malvino and David J Bates, 7th Edition, Tata McGraw –Hill.
2. Electronic Devices and Circuits Theory, Boyelstad, Pearson Education, 8th Edition, September 2011.
3. Op-Amps and Linear Integrated Circuits , - Ramakanth A. Gayakwad, PHI, 4th Edition, 2009
4. Linear Integrated Circuits – D. Roy Chowdhury, New Age International Pvt.Ltd., 2nd Edition, 2003.

References:

1. Electronic Devices and Circuit Theory, Robert L.Boylestad and Louis Nashelsky, 10th Edition(2010), Pearson/PHI
2. Electronic Devices and Circuits, David A.Bell, Oxford, 5th edition, 2009.
3. Electronic Devices and Circuits, S.Salivahanan, Kumar, Vallavaraj, TATA McGraw Hill, 2nd Edition,2003.
4. Operational Amplifiers & Linear ICs, David A Bell, Oxford Uni. Press, 3rd Edition, 2005.

e-learning resources:

http://nptel.ac.in/courses.php

http://jntuk-coeerd.in/

C PROGRAMMING(Required)
(Common to ECE, AE during I B.Tech., I Semester)
(Common to CSE, EEE, CE, ME, IT during I B.Tech., II Semester)
Course Code(s): AE1T4, EC1T5, EE2T6, CE2T6, ME2T6, CS2T6, IT2T6

Credits: 3

Lecture: 3 periods/week **Internal assessment: 30 marks**
Tutorial: 1 period /week **Semester end examination: 70 marks**

Prerequisites: Nil

Course Objectives:

- Learn the structure, syntax and semantics of C programming.
- Learn different control structures like decision control, loop control and arrays.
- Learn the modular programming concepts and storage classes.
- Learn the limitations of basic data types and learn the concepts of derived data types and user defined data types.
- Learn how to perform various FILEI/O.

Course Outcomes:

At the end of this course student will:
CO1) Understand the structure, syntax and semantics of C programming
CO2) Choose different control structures like decision control, loop control to solve the problem
CO3) Study the modular programming concepts and storage classes
CO4) Learn the concepts of basic data types, derived data types and user defined data types
CO5) Understand how to perform various FILE I/O operations

Course Contents / Syllabus

UNIT –I

Topic Level Objective: Notion of Computer Languages, algorithm, computational procedure, editing and executing programs and C Declarations.

BASICS AND INTRODUCTION TO C:

Basics of Computer, Introduction to C, About ANSI C Standard, Machine, Assembly and High-level Language, Assembler, Compiler and Interpreter, Structure of a C program, Programming Rules, Executing the C Program, Standard Directories, Advantages of C, Header Files, Flow Chart, Algorithm, Analyzing Algorithm, Classification Algorithms.

THE C DECLARATIONS: The C-Character set, Delimiters, Types of Tokens, The C keywords, Identifiers, Constants, Variables, C Data types, dynamic initialization, type modifiers, type conversions, constant and volatile variables. Properties of Operators,
Operator Priority ,comma and conditional operators, arithmetic, relational, assignment operators and expressions, logical , bitwise operators. Input and output in c: Formatted and Unformatted functions.

UNIT-II

Topic Level Objective: Understanding branching, iteration, data representation using arrays and strings.

DECISION STATEMENTS:

The if statement, if-else, nested if else, if-else-if ladder, break, continue, goto, Switch statement, nested switch case, Switch case and nested ifs.

LOOP CONTROL: for loop, nested for loop, while, do-while, do-while statement with while loop.

ARRAYS:

Array initialization, array terminology, characteristics of an array, 1-D array and its operations, predefined streams, 2-D arrays and operations, Multi -dimensional arrays.

STRINGS: Declaration and initialization of string, string standard functions, string conversion functions, memory functions, application of strings.

UNIT-III

Topic Level Objective:Modular programming and recursive solution formulation and storage classes.

FUNCTIONS:

Basics, function definition, return statement, types of functions, call by value ,call by reference, function as an argument, Functions with operators, Function and Decision Statements, Functions and loop Statements, Functions with arrays and Pointers,
Recursion-Types of Recursion, Rules for Recursive Function, Recursion versus
Iterations, Advantages and Disadvantages of Recursion, Efficiency of Recursion, Library Functions.

STORAGE CLASS: Variable Lifetime, Automatic Variables, External Variables, Static Variables, Register Variables.

UNIT-IV:

Topic Level Objective:Understanding pointers, dynamic memory allocation and Preprocessor Directives.

POINTERS:

Features of pointers, pointers and address, pointer declaration, void pointers, arithmetic operations with pointers, pointers and arrays, array of pointers, pointers to pointers,pointers and strings. Dynamic memory allocation, memory models, memory allocation functions.

PREPROCESSOR DIRECTIVES:

The #define Directive, Undefining a Macro, Token Pasting and Stringizing Operators, The #include Directive, Conditional Compilation, The Predefined Macros in ANSI and Turbo-C,Standard I/O Predefined Streams in stdio.h, The Predefined Macros in ctype.h.

UNIT V:

Topic Level Objective: Understanding derived data types of C and basic of file operations.

STRUCTURE AND UNION: Features of Structures, Declaration and initialization of Structures, Structure within Structure, Arrays of Structure, Pointer to Structure,

Structure and functions,typedef,Bit fields, Enumerated Data Type, Union, Union of Structures.

FILES:

Streams and File Types, Steps for File Operations, FILE I/O, Structures Read and Write, Other file function, Command line Arguments, Application of command line arguments, Environment variables.

Text book:

1.Programming in C, by Ashok N.Kamthane, (2nd edition), Pearson publications,2011.

Reference books :

1. Programming in ANSI C (5th Edition) by E.Balaguruswamy, McGraw-Hill publications.

2. “A first book of ANSI C”, 3rd edition, by Gray J .Brosin, cengagedelmar Learning India P.ltd publications.

3. Problem Solving with C by M.T Somashekara PHI publications.

4. C Programming Language”, (2nd edition) by Brain W.Kernighan & Dennis Ritchie, “, PHI publication

E-learning resources:

http://nptel.ac.in/courses.php

http://jntuk-coeerd.in/

ENGNEERING PHYSICS & CHEMISTRY LAB
(Only for ECE during I B.Tech., I Semester)
(Common to CSE, IT, EEE during I B.Tech., II Semester)

Course Code(s): EC1L2, CS2L1, IT2L1, EE2L1

Credits: 2

Lecture: --- **Internal assessment: 25 marks**

Lab Hours: 3 period /week **Semester end examination: 50 marks**

Prerequisites: Engineering Physics & Chemistry Lab

COURSE OBJECTIVES:

To make student

- Knowledgeable in different concepts of physics such as Properties of Matter, Sound, Electricity, Optics and Electronics by explaining through experiments.
- Familiar with quality and parameters of water samples, useful for drinking effluent treatment and agriculture purposes.
- Aware of preparation of some plastic material and corrosion kinetics useful in industries.
- Know about the measuring the properties of the lubricants which are industrially useful.

COURSE OUTCOMES:

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

CO1)Determine the physics parameters such as rigidity modulus, refractive index, velocity of sound,time constant, magnetic induction,radius of curvature of a lens,enegy gap, breakdown voltage of zener, wavelength of a monochromatic source, Numerical aperture and Attenuation in optical fibre.

CO2)Learn hardness , alkalinity, turbidity ,D.O of water sample students can understand different methods of water treatment.

CO3)know nature of the soil from PH values the types of fertilizers and pesticides to be used can be decided

CO4)know the purity of water which is useful for drinking and industrial purposes.

CO5)know the preparation of Bakelite and understand to applications in industries.

ENGINEERING PHYSICS

ANY SIX OF THE FOLLOWING:

MECHANICS:

1)Determine the rigidity modulus of the material of the wire using torsional Pendulum

SOUND:

2) Determine the velocity of sound by volume resonator method.

OPTICS:

1. Determine the wavelength of a source by normal incidence method using diffraction grating

2. Determine the radius of curvature of a plano convex lens by forming Newton's Rings

3. Determine the refractive index of the material of the prism (minimum deviation method) using spectrometer.

ELECTRICITY AND MAGNETISM:

1. Study the variation of magnetic field along the axis of a solenoid coil using Stewat – Gee's apparatus.

2. Determine the time constant for a C-R circuit.

ELECTRONICS:

1. Study of characteristic curves of a zener diode to determine its break down Voltage

2. Determine band gap of semiconductor using a p-n junction diode.

3. Draw the characteristic curves and determine thermoelectric coefficient of a Thermistor.
4. Determine the Numerical Aperture of an optical fibre.

5. Determine the attenuation in the optical fibre.

ENGINEERING CHEMISTRY

1. Determination of Total Hardness of water sample using EDTA.

2. Determination of Total alkalinity of water sample.

3. Determination of D.O in water.

4. Measurement of Turbidity of water sample.

5. pH of Soil and fruits.

6. Preparation of Phenol-Formaldehyde resin.

BASIC ELECTRONICS ENGINNERING LAB
(Common to CSE, IT during I B.Tech., II Semester)

Course Code: CS2L2, IT2L2 **Credits: 2**
Lecture: -- **Internal assessment: 25 marks**
Lab: 3 period /week **Semester end examination: 50 marks**

Course Objectives:

2
1. To study basic electronic components.
2. To observe characteristics of electronic devices.
3. To get the practical exposer of the Op-amp applications.
4. To study the practical limitations of the Op-amp.
5. To study Linear & Non linear wave shaping.

Course Outcomes:

The student will be able

CO1)To apply the concepts and analytical principles to analyze electronic (diodes, transistors) circuits.

CO2) To Understanding of the operation diodes , rectifiers and transistors in order to build circuits.

CO3)To learn to the characteristics of Transistor.

CO4) To learn the basics of Amplifiers.

CO5) The students are able to design Op-amp circuits.

Part I

1. The identification & Testing of Electronic component like R,L,C, Diodes,Transistors etc.
2. Study of CRO, function generator, regulated power supply etc.,

Part II any TEN Experiments

1. Diode Characteristics(S)
   a) Forward Bias b) Reverse Bias
2. Zener Diode Characteristics
3. Half Wave rectifier with & without filter
4. Full Wave rectifier with & without filter
5. Transistor CB Characteristics (I/P & O/P)
6. Transistor CE Characteristics (I/P & O/P)
7. CE Amplifier
8. CC Amplifier
9. CB Amplifier
10. Op-amp inverting amplifiers (OP -AMP Applications) – Adder, Subtractor, Comparator Circuits.
11. Op-amp non-inverting amplifiers(OP -AMP Applications)–Adder,Subtractor,Comparator Circuits
12. Op-amp inverting amplifier for desired gain and bandwidth.

Learning Resources

REFERENCE BOOKS:

1. Electronic devices & circuits by B.L.Theraja,R.S.Sedha,S.Chand publications
2. Electronic devices & circuits by Robert L.Boylested
3. Linear Integrated Circuits – D. Roy Chowdhury, New Age International Pvt.Ltd., 2nd Edition, 2003.

C PROGRAMMING LAB

(Common to ECE, AE during I B.Tech., I Semester)

(Common to EEE, CE, ME, CSE, IT during I B.Tech., II Semester)

Course Code(s) : AE1L2, EC1L3, CE2L3, ME2L3, CS2L3, IT2L3, EE2L3

Credits: 2

Lecture: - **Internal assessment: 25 marks**

Lab Hours: 3 periods /week **Semester end examination: 50 marks**

Prequisites: C Programming

Course Objectives:

1. To make the student learn a programming language.
2. To learn problem solving techniques.
3. To teach the student to write programs in C and to solve the problems.

Course Outcomes:

At the end of this course student will:

CO1) Implement simple program including operators.

CO2) Choose and Implement Programs using control structures.

CO3) Develop programs using user defined functions.

CO4) Develop programs using derived data types, user defined data types.

CO5) Implement programs to perform various file operations.

Course Contents / Syllabus

Exercise1: Basics

1. Write a program to print sample strings like “hello world”, “Welcome to C Programming” with different formats using escape sequences.
2. Write a Program to print different data types in ‘C’ and their ranges.
3. Write a Program to initialize, assignment & printing variables of different data types.

Exercise2: Operators

4. Write a Program to demonstrate arithmetic operators. (+,-,*,/,%)

5. Write a Program to demonstrate logical operators.(logical AND, logical OR)
6. Write a Program to read radius value from the keyboard and calculate the area of circle and print the result in both floating and exponential notation.
7. Write a Program to calculate simple interest.
8. Write a Program to convert temperature. (Fahrenheit –Centigrade and vice-versa)

Exercise3: Operators

1. Write a Program to demonstrate relational operators.(<,>,<=,>=,==,!=)
2. Write a program to check equivalence of two numbers using conditional operator.
3. Write a Program to demonstrate pre increment and post increment.(++a, a++ where a is a value to be initialized)
4. Write a Program to demonstrate pre decrement and post decrement.(--a, a--where a is a value to be initialized)
5. Write a program for computing the volume of sphere, cone and cylinder assume that dimensions are integer's use type casting where ever necessary.

Exercise4: Decision Statements

4. Write a Program to read marks of a student in six subjects and print whether pass or fail (using if-else).
5. Write a Program to calculate roots of quadratic equation (using if-else).
6. Write a Program to calculate electricity bill. Read starting and ending meter reading. The charges are as follows.

No. of Units Consumed	Rate in(Rs)
1-100	1.50 per unit
101-300	2.00 per unit for excess of 100 units
301-500	2.50 per unit for excess of 300 units
501-above	3.25 per unit for excess of 500 units

Exercise5: Switch operations

1. Write a Program to perform arithmetic operations using switch case.
2. Write a Program to display colors using switch case (VIBGYOR).
3. Write a Program to display vowels and consonants using switch case.
4. Write a Program to display names of days in a Week using switch case.

Exercise6: Basic Loop operations

Do the Following Programs Using for, while, do-while loops.

1. Write a program to calculate sum of individual digits of a given number.
2. Write a program to check whether given number is palindrome or not.
3. Write a program to print prime numbers in the given range.
4. Write a program to display multiplication tables from 1 to 10 except 3 and 5.

Exercise7: Advanced loops

1. Write a program to print the Fibonacci series for given 'N' value.
2. Write a program to check whether a given number is a Fibonacci number or not.
3. Write a program to read 2 numbers x and n then compute the sum of the Geometric Progression. $1+x+x^2+x^3+ - - - - - - - +x^n$
4. Write a program to print the following formats.

```
1                    *
1 2                  * *
1 2 3                * * *
1 2     3 4          * * * *
```

Exercise8: 1-D arrays

1. Write a program to store 10 elements in the 1-D array and print sum of the array.
2. Write a program to print minimum and maximum elements in the 1-D array.
3. Write a program to count no. of positive numbers, negative numbers and zeros in the array.
4. Write a program to search the given element by using linear search.
5. Write a program to sort the given elements using bubble sort technique.

Exercise9: 2-D arrays

1. Write a program to perform matrix addition and matrix subtraction.
2. Write a program to perform matrix multiplication by checking the compatibility.
3. Write a program to print the transpose of a matrix.

Exercise10: Strings

1. Write a program to perform various string manipulations using built-in functions.
2. Write a program to print the given strings in ascending order.

3. Write a program to verify the given string is palindrome or not (without built-in functions, with using built-in functions).
4. Write a program to concatenate two strings using arrays.

Exercise 11: Math Functions and I/O Fucntions

1. Write a program to read values from keyboard and find the values using abs(),sqrt(),floor(),ceil()and pow().
2. Write a program to read and display a value using getch() and putch().
3. Write a program to read and display a value using getchar(), putchar(),gets() and puts().

Exercise 12: Functions

1. Write a program to find sum of two numbers using functions.
2. Write a program to find product of two numbers using functions without arguments, without return type.
3. Write a program to find difference of two numbers using functions without arguments, with return type.
4. Write a program to find sum of two numbers using functions with arguments &without return type.
5. Write a program to find product of two numbers using functions with arguments, with return type.

Exercise13: Functions and Recursion

1. Write a program to swap two numbers using

a) Call By Value B) Call By Reference.

2. Write a program to calculate factorial, gcd using recursion and non-recursion functions.

1. Write program to perform arithmetic operations using pointer.
2. Write a program matrix addition using pointers.

Exercise14: Structures

1. Write a program to create structure for an account holder in a bank with following Fields: name, account number, address, balance and display the details of five account holders.

2. Write a program to find total marks of individual student and average marks for 10 students using structures.
3. Write a program to create structure called traveler and members of structure are train no, coach no, seat no, source ,destination , gender, age, name and departure date.

4. Write a program to illustrate passing an entire structure to a function.

Exercise15: File operations using command line arguments

1. Write a program which copies the contents of one file to another file using command line arguments.
2. Write a program to reverse the first n characters in a file use command line arguments.

Reference Books :

1 .Problem Solving and Program Design in C, 4th edition, by jeri R. Hanly and Elli B.Koffman.
2.Programming in C by Pradip Dey, Manas Ghosh 2nd edition Oxford University Press.
3. E.Balaguruswamy, Programming in ANSI C 5th Edition McGraw-Hill
4. A first book of ANSI C by Gray J.Brosin 3rd edition Cengagedelmer Learning India P.Ltd
5.AL Kelly, Iraphol,Programming in C,4th edition Addison-Wesley – Professional
6.Brain W.Kernighan & Dennis Ritchie, C Programming Language, 2nd edition, PHI

II/IV B. TECH. FIRST SEMESTER
DISCRETE MATHEMATICS
(Required)

Course Code : CS 3T1 **Credits: 3**
Lecture: 3 periods/week **Internal assessment: 30 Marks**
Tutorial: 1period/week **Semester end examination: 70 Marks**

Prerequisites: C Programming

Course Objectives:

1. To know the notations used in the discrete mathematics associated with computer science and engineering.
2. To learn the rudiments of elementary mathematical reasoning (elementary proofs; proofs by induction, Normal forms)
3. To understand the theoretical parts of all further courses in Computer Sciences.
4. To understand the fundamentals of counting and discrete probability
5. To understand basic set-theoretical notions: relations, functions, graphs, equivalence relations, and orderings.

Course Outcomes:
At the end of this course student will:

CO1) Apply fundamentals of mathematical logic for proof techniques

CO2) Understand the concepts of partial ordering

CO3) Illustrate various types of trees and their applications

CO4) Demonstrate various types of graphs and its applications

Syllabus:

UNIT I

Statements and Notation, Connectives- Negation, Conjunction, Disjunction, Conditional and

Bi-conditional, Statement formulas and Truth Tables. Well formed formulas, Tautologies, equivalence of formulas, Duality Law, Tautological Implications, Functionally Complete Sets of Connectives, Other connectives.

UNIT II
Inference calculus - Derivation process - Conditional proof - Indirect method of proof- Automatic theorem proving - Predicate calculus.

UNIT III

Partial ordering – Lattices – Properties - Lattices as algebraic system - sub lattices - Direct product and homomorphism - Special lattices - Complemented and Distributive lattices.

UNIT IV

Graphs – Baisc Concepts – isomorphism-sub graphs –Trees and Their Properties – Spanning Trees – Directed Trees – Binary Trees

UNIT V

Planar Graphs – Euler Graphs – Multigraphs and Euler Circuits – Hamiltonian Graphs – Chromatic Numbers – The Four – Colour Problem.

Learning Resource

Text Books

1. J P Trembly and R Manohar , Discrete Mathematical Structures with Applications to Computer Science. TMH
2. Joe L. Mott. Abraham Kandel and Theodore P.Baker, Discrete Mathematics for Computer Scientists & Mathematicians. PHI,Second Edition

References

1. Kenneth H. Rosen, "Discrete Mathematics and its Applications", Tata McGraw-Hill Publishing Company, Pvt. Ltd., Fifth edition, New Delhi, 2003. 2. C.L. Liu, "Elements of Discrete Mathematics", Second edition, McGraw-Hill Book Company, New York, 1988.

II/IV B. TECH. FIRST SEMESTER

DATA STRUCTURES
(Required)

Course Code : CS 3T2 **Credits: 3**
Lecture: 3 periods/week **Internal assessment: 30 Marks**
Tutorial: 1period/week **Semester end examination: 70 Marks**

Prerequisites: C Programming

Course Objectives:

1. Allow to assess how the choice of data structures and algorithm design methods impacts the performance of programs
2. To choose the appropriate data structure and algorithm design method for a specified application.
3. To solve problems using data structures such as linear lists, stacks, queues, binary trees, binary search trees, and graphs and writing programs for these solutions.
4. To efficiently implement the different data structures and solutions for specific problems.

Course Outcomes:

At the end of this course student will:

CO1) Analyze the concepts of algorithm evaluation and find time and space complexities for searching and sorting algorithms.

CO2) Implement linear data structure such as stacks, queues, linked lists and their applications.

CO3) Implement basic operations on binary trees

CO4) Demonstrate the representation and traversal techniques of graphs and their applications

Syllabus:

UNIT 1

Introduction, searching and sorting: *Algorithm specification*: Introduction, Recursive algorithms, Data Abstraction, Performance Analysis: Space complexity, time complexity, asymptotic notation, *Searching*: Linear and Binary search algorithms, *Sorting*: Bubble sort, Selection sort, Insertion sort, quick sort, merge sort

UNIT 2

Stacks and Queues: Stacks, stacks using dynamic arrays, queues, circular queues using dynamic arrays, Evaluation of an expression: Expressions, evaluating postfix expression, conversion of infix expression to postfix expression.

UNIT 3

Linked Lists: Single linked lists, Representing chains, operations for chains, operations for circularly linked lists, doubly linked lists, Polynomials: Representation, adding polynomials, sparse matrix representation, linked stacks and queues

UNIT 4

Trees: Introduction: Terminology, representation of trees, binary trees: abstract data type, Properties of binary trees, binary tree representation, binary tree traversals: Inorder, preorder, postorder, Binary search trees: Definition, searching BST, insert into BST, delete from a BST, Height of a BST

UNIT 5

The Graph ADT: Introduction, definition, graph representation, elementary graph operations: BFS, DFS, Spanning trees, minimum cost spanning tree: Prim's, Kruskal's algorithms.

Learning Resource

Text Books

1. Fundamental of Data Structures in C – 2nd Edition, Horowitz, Sahani, Anderson-Freed, University Press.

References

1. Data Structures and Algorithm Analysis in C – 2nd Edition, Mark Allen Weiss, Pearson
2. Classic Data Structures – 2nd Edition, Debasis Samantha, PHI.

II/IV B. TECH. FIRST SEMESTER
PROGRAM DESIGN
(Required)

Course Code : CS 3T3	**Credits: 3**
Lecture: 3 periods/ week	**Internal assessment: 30 Marks**
Tutorial: 1period/week	**Semester end examination: 70 Marks**

Prerequisites: C Programming

Course Objectives:

1. Problem Solving and Program design in C teaches a disciplined approach to problem solving and to applying widely accepted software engineering methods to design program solutions as coupling, cohesion, modular programming and debugging.

Course Outcomes:

At the end of this course student will:

CO1) Demonstrate the step by step notation to solve simple mathematical and logical problems

CO2) Implement the 'C' programs for solutions of mathematical and logical problems

CO3) Apply code reading and debugging techniques to analyze and interpret and describe the purpose of program code

CO4) Apply the modular programming techniques to simplify the programs

CO5) Implement the simple programs using derived and user defined data types to organize the date items

Syllabus:

UNIT 1

Top-down Design with Functions: Building programs from existing information- case studies, library functions, top down design and structure charts - case study, functions without arguments, functions with input arguments, Case Study.

UNIT 2

Functions with simple output parameters, Multiple calls to a function with input/output parameters, Scope of names, Formal output parameters as Actual Arguments, A program with multiple functions-Case study, Debugging and testing a program system , Recursion: The nature of recursion, Tracing a recursive function, Recursive Mathematical functions

UNIT 3

Array arguments, Parallel Arrays and Enumerated Types, Multidimensional arrays, Strings: string library functions, Array of pointers, Structures and Union types: User-defined structure types, structure type data as input and output parameters, functions whose result values are structured, Problem Solving with Structured Types, Case Study, Introduction to Union.

UNIT 4

Pointers, Dynamic Memory Allocation, Text input/output files, Binary files, searching a database- Case study

UNIT 5

Programming in the Large : Using Abstraction to Manage Complexity, Personal Libraries: Header Files, Implementation Files, Storage Classes, Modifying functions for inclusion in a Library, Arguments to Function Main, Defining Macros with Parameters, Common Programming Errors.

Learning Resource

Text Books

1. Problem Solving and Program Design in C, Jeri R. Hanly, Ellot B. Koffman, 7th Edition, Pearson.

References

1. Programming in C, Pradip Dey, Manas Ghosh, 2nd Edition, Oxford University Press.
2. How to Solve it by Computer- R.G.Dromey, PHI.
3. A First Book of ANSI C, Gary J.Bronson, 3rd Edition, Cengage.
4. A Book on C, AL KELLY and IRA POHL, 4th Edition, Pearson.
5. The C Programming Language, Brain W.Kernighan & Dennis Ritchie, 2nd Edition, PHI.

II/IV B. TECH. FIRST SEMESTER
FORMAL LANGUAGE AUTOMATA THEORY (Required)

Course Code : CS 3T4 **Credits: 3**
Lecture: 3 periods/ week **Internal assessment: 30 Marks**
Tutorial: 1period/week **Semester end examination: 70 Marks**

Prerequisites: Discrete Mathematics, Regular Languages and Finite Automata & Mathematics for Computation Theory

Course Objectives:

1. To understand the fundamental models of computation
2. To determine Chomsky classification of languages.
3. To classify machines by their power to recognize language
4. Understand the concepts of decidability, NP-Completeness and NP Hard Problems

Course Outcomes:

At the end of this course student will:

CO1) Analyze and design Finite Automata

CO2) Classify the devices according to their computational power

CO3) Understand the concept of the Formal grammars and languages

CO4) Understand Turing machine concept and the techniques applied in computers

CO5) Understand basic complexity classes like P & NP

Syllabus:

UNIT 1
Fundamentals: Strings, Alphabet, Language, Operations, Chomsky hierarchy of languages
Finite state machine: Definitions, finite automation model, acceptance of strings and languages, DFA and NFA, transition diagrams and language recognizers. NFA with ε transitions – Equivalence between NFA with and without ε transitions, NFA to DFA conversion, minimization FSM, equivalence between two FSM's, Output machines- Moore and Mealy machine.

UNIT 2

Regular Languages : Regular Sets , Regular Expressions , identity Rules, Constructing Finite automata for a given regular expressions, Conversion of Finite automata to regular expressions, Pumping lemma of regular sets , closure properties of regular sets (proofs not required).**Regular Grammars** – right linear and left linear grammars, equivalence between regular grammar and FA.

UNIT 3

Context Free Grammar: derivation trees, sentential forms, right most and left most derivations of strings, Ambiguity in Context free Grammars, Minimization of Context free grammars, CNF, GNF, Pumping Lemma for Context Free Languages. Enumeration of properties of CFL (proofs omitted).**Push down Automata:** definition, model, acceptance of CFL, Acceptance by final state and acceptance by empty state and its equivalence

UNIT 4

Equivalence of CFL and PDA (proofs not required), Introduction to DCFL and DPDA.**Turing Machine:** Definition, model, Design of TM, computable functions, recursively enumerable languages. Church's hypothesis, counter machine, types of Turing Machines (proofs not required)

UNIT 5

Computability Theory: Decidability of problems, Universal TM, Un decidable problems about Turing Machine – Post's Correspondence Problem - The classes P and NP.

Learning Resource

Text Books

1. Formal Languages and Automata Theory by Basavarj S. Anami,Karibasappa K.G, WILEY-INDIA
2. H.E.Hopcroft, R.Motwani and J.D Ullman, "Introduction to Automata Theory, Languages and Computations", Second Edition, Pearson Education, 2003.

References

1. Theory of Computer Science, Automata languages and computation, 2/e,Mishra,Chandra Shekaran, PHI

2. H.R.Lewis and C.H.Papadimitriou, "Elements of The theory of Computation", Second Edition, Pearson Education/PHI, 2003

3. J.C.Martin, "Introduction to Languages and the Theory of Computation", Third Edition, TMH, 2003.
4. Michael Sipser, "Introduction of the Theory and Computation", Thomson Brokecole, 1997.

II/IV B. TECH. FIRST SEMESTER

OBJECT ORIENTED PROGRAMMING THROUGH JAVA (Required)

Course Code : CS 3T5	**Credits: 3**
Lecture: 3 periods/ week	**Internal assessment: 30 Marks**
Tutorial: 1period/week	**Semester end examination: 70 Marks**

Prerequisites: C Programming

Course Objectives:

1. The main objective of this course is to understand the Object Oriented programming issues in developing more complex software designs. Students will also learn the advantages of Object Oriented programming over the normal and old paradigm structured programming languages. Examples which are demonstrated using java helps the students to understand the concepts and apply the features of Object Oriented programming. The enhancements that are made in the latest certification exams for java are also kept in view. This helps students to keep their skills up to date.

Course Outcomes:

At the end of this course student will:

CO1) Understand the key features of the Java programming language

CO2) Apply essential object-oriented programming concepts like dynamic polymorphism, abstract (virtual) methods using Java

CO3) Apply the principles behind good object-oriented design.

CO4) Get exposure to the latest trends in java language and its compatibility in handling numerous complex domains.

Syllabus:

UNIT 1

Java Basics and Anatomy:

Java Basics: OOP's principles, Java History, advantages, Data types, operators, expressions, control statements, methods and recursion, sample programs.

Java Anatomy: Java Objects and References, Constructors, this keyword, Arrays (single and multi-dimensional), String and its immutability, Buffer &Builder Classes, String Tokenizer

UNIT 2

Inheritance (Extending and Implementing): Introduction, Derived Classes, Advantages and Types of Inheritance, Implementation, Inheritance and Member Accessibility. Overriding, Super, Abstract classes and Methods, Final Classes and Final Methods, Dynamic Binding, Polymorphism.

Interfaces: differences between classes and interfaces, defining an interface, implementing interface, variables in interface, extending interfaces.

UNIT 3

Packaging and Java API

Packages: Defining, Creating and Accessing a Package, importing packages, access controls (public, protected, default, and private). Wrapper Classes and Auto Boxing, I/O classes

Collections Framework: Object class, importance of methods like hash code ( ) and equals ( ). Array List, application of Comparable and Comparator interfaces.

UNIT 4

Exception handling and Multithreading:

Concepts of exception handling, benefits of exception handling, usage of try, catch, throw, throws and finally, built in exceptions, creating own exception. Threads: Thread life cycle, creating threads, synchronizing and Communication of threads.

UNIT 5

Graphical User Interaction

Graphical User Interaction: Swings- Introduction, limitations of AWT, components, containers, exploring swing-Frame and JComponent, Icons and Labels, text fields.

Layout managers– border, grid, flow. Event Handling: Events, Event Delegation Model, Event classes, Listeners, handling mouse and keyboard events.

Learning Resource

Text Books

1. Java Fundamentals, a Comprehensive Introduction, Herbert Schildt & Dale Skrien, 2013, McGraw-Hill.

References

1. Introduction to Java Programming 7/e, Brief version, Y.Daniel Liang, Pearson
2. Thinking in Java 4E: Bruce Eckel, Pearson
3. The Java™ Programming Language: Ken Arnold, James Gosling, Pearson.

II/IV B. TECH. FIRST SEMESTER
DATA STRUCTURES LAB (Required)

Course Code : CS 3L1 **Credits: 2**
Lab Hours: 3 periods/ week **Internal assessment: 25 Marks**
Tutorial:- **Semester end examination: 50 Marks**

Prerequisites: Data Structures

Course Objectives:

1. To implement recursive functions.
2. To arrange data using different sorting techniques.
3. To implement stack, queue, linked list, tree and graph data structures.

Course Outcomes:
At the end of this course student will:

CO1) Implement different sorting and searching algorithms

CO2) Implement the stack, Queue and their applications

CO3) Implement various types of linked lists and their applications

CO4) Perform basic operations on trees and graphs and determine minimum spanning tree

Syllabus:

Implement the following exercises using 'C' Programming language.

Exercise 1

1. Write recursive program which computes the n^{th} Fibonacci number, for appropriate values of n.
2. Write recursive program for calculation of Factorial of an integer.
3. Write a program to search an element using linear and binary search with and without recursion.

Exercise 2

90, 77, 60, 99, 55, 88, 66, 32, 41, 19

1. Arrange above data set using Bubble sort
2. Arrange above data set using Selection sort

Exercise 3

26, 5, 77, 1 , 61, 11, 59, 15, 48, 19

1. Arrange above data set using insertion sort
2. Arrange above data set using Quick sort
3. Arrange above data set using Merge sort

Exercise 4

1. Implementation of stack operations using static and dynamic arrays.
2. Implementation of queue operations using static and dynamic arrays.

Exercise 5

1. Railroad cars numbered are as 0,1,2,---,n-1. Each car is brought into the stack and removed at any time. For instance, if n=3, we could move 0, move 1, move 2 and then take the cars out, producing 2,1,0. Implement application for the given problem.

2. Consider a payment counter at which the customer pays for the items purchased. Every time a customer finished paying for their items, he/she leaves the queue from the front. Every time another customer enters the line to wait, they join the end of the line. Implement the application for this problem.

Exercise 6

Implementation of single linked list.

Exercise 7

Implementation of doubly linked list

Exercise 8

3. Representation of Sparse matrix.
4. Implementation of circular linked list

Exercise 9

Implement Exercise 5 (a) using linked lists.

Exercise 10

Implement Exercise 5(b) using linked lists.

Exercise 11

A polynomial has the main fields as coefficient, exponent in linked list it will have one more field called link to point to next term in the polynomial. If there are n terms in the polynomial then n such nodes has to be created.

Exercise 12

Implementation of binary tree: creation, insertion, deletion, traversing

Exercise 13

Implementation of Binary Search Tree operations

Exercise 14

Implementation of Graph traversals

Exercise 15

Implementation of minimum spanning tree

II/IV B. TECH. FIRST SEMESTER
ADVANCED C PROGRAMMING LAB
(Required)

Course Code : CS 3L2 **Credits: 2**
Lab Hours: 3 periods/ week **Internal assessment: 25 Marks**
Tutorial:- **Semester end examination: 50 Marks**

Prerequisites: C Language

Course Objectives:

1. The objective of this course is to introduce advanced constructs in C programming language. Students will apply these constructs and problem solving methodology to solve complex problems

Course Outcomes:

At the end of this course student will:

CO1) Analyze and apply the debugging techniques in 'C' programs.

CO2) Develop 'C' programs in structured programming orientation.

CO3) Develop the user defined data types for simplifying the data manipulation.

CO4) Apply the file manipulation concepts to organize data.

CO5) Implement the 'C' programs by a team of people for engineering applications

Syllabus:

- Implement 'C' Program for the following.

Exercise – 1

1. Pass the radius as parameter to a function, Compute and display area and circumference of a circle.
2. You want to draw a diagram (House) on screen and prepare a structure chart.
3. Implement a function to perform addition and subtraction of two complex numbers and prepare structure chart.

Exercise - 2

1. Develop a Function Scale multiplies its first argument by 10 raised to the power indicated by its second argument (an integer).
2. Develop a function that separate the given floating point number into sign, whole number magnitude and fractional part.
3. Implement a function to sort three numbers using call by reference.

Exercise - 3

1. You are working problems in which you must display your results as integer ratios. Therefore you need to be able to perform computations with common fractions and get results that are common fractions in reduced form. You want to write a program that will allow you to add, subtract, multiply, and divide several pairs of common fractions.
2. Write a recursive function find_sum that calculates the sum of successive integers starting at 1 and ending at n (i.e., find_sum(n)=1+2+3+…..(n-1)+n).

Exercise - 4

1. Write a program to print a table of differences. Compute the mean and standard deviation of an array of data and displays the difference between each value and the mean.
2. Using arrays to represents the three vertices of a triangle, calculate the length of the three sides of triangle formed with these vertices. Then determine whether triangle can be formed with given vertices.

Exercise - 5

1. The Sales manager of your company needs a sales analysis program to track sales performance by salesperson and by quarter. The program will read all sales transactions from a two dimensional array. The data for each transaction will be the salesperson's number, the quarter in which sale took place, and the sales amount. The sales transactions are in no particular order. After scanning all sales transactions, the program should display a table form which includes totals by person and by quarter.

Exercise - 6

1. Implement a function to change the representation of a date from a string containing day, month and name and year to three integers(month day year) and vice versa.
2. Write a program to delete the given word in a sentence.

   Example : Enter the sentence : Ram is a good boy

   Enter the word to be deleted : good

   After deletion the sentence is as follows ….. Ram is a boy.

Exercise - 7

1. A company manufactures three types of UPS models. At the end of any month serial numbers are to be generated for each of the models manufactured. The last serial number of each type of model has to be taken into consideration for generating the new set of serial number for each set of models. At the end of particular month the following data is available for generating the new serial numbers model-wise:

Model Type	Units manufactured	Last serial number
Ups1	3	24
Ups2	4	19
Ups3	5	9

A program has to be written that generates the new serial numbers for each model type for the month under consideration taking the above as input.

Exercise - 8

1. We are working on engineering project that use complex number for modeling of electric circuits. We need to develop a user-defined structure type and a set of operations that will make complex arithmetic virtually as straight forward as arithmetic on C's built-in numeric types.

Exercise - 9

1. Write a program to read and display a two dimensional array by reading number of rows and columns, And also allocate memory dynamically for an array.
2. The CEO signs vouchers, cheques, and documents where the amount is given in digits as well as in words. Every time before signing these, the CEO checks up whether the amount written in words matches with that of digits. To do this swiftly the CEO needs a program in the counter that would accept the value written on vouchers, cheques, and documents and display the amount in words. The amount in any case should not exceed Rs.99 crores. A program has to written to perform this task.

Exercise - 10

1. A program has to be written that takes student data as input and display the same in the order of entry. It should provide the user the option to choose to display of a particular student given by their name or the roll number.

   Displaying all student records in an order of name or ascending order of grade should also be provided as options.

Exercise - 11

1. Write a C program that takes the name of a file as a command line argument , opens the file, reads through it to determine the number of words in each sentence, displays the total number of words and sentences, and computes the average number of words per sentence. The results should be printed in a table such as below:

   This program counts the words and sentences in file " comp.txt".

   Sentence : 1 Words : 29

   Sentence : 2 Words :41

   Sentence : 3 Words : 16

   File "comp.txt" contain 86 words in 3 sentences for an average of 28.3 words per sentence.

Exercise - 12

1. A computer phone book containing name of persons, their home, office and mobile phone numbers has to be prepared. This book should have the necessary provision for adding, editing and deleting phone numbers and display the phone number of any person from the Phone book as and when required.

Learning Resource

Text Books

1. Problem Solving and Program Design in C, Jeri R. Hanly, Ellot B. Koffman, 7th Edition, Pearson.

2. Programming in C , Pradip Dey, Manas Ghosh, Second Edition , Oxford Education

II/IV B. TECH. FIRST SEMESTER
JAVA LAB(Required)

Course Code : CS 3L3	**Credits: 2**
Lab Hours: 3 periods/ week	**Internal assessment: 25 Marks**
Tutorial:-	**Semester end examination: 50 Marks**

Prerequisites: Object Oriented Programming Through Java

Course Objectives:

Practical implementations based on the OOPs features using JAVA.

Course Outcomes:

At the end of this course student will:

CO1) Apply the key features of the Java programming language

CO2) Apply essential object-oriented programming concepts

CO3) Apply object-oriented programming techniques like dynamic polymorphism, abstract (virtual) methods using Java

CO4) Design an application with strong and good object-oriented design principles.

CO5) Use various IDE's to implement Java Programs.

Syllabus:

Exercise-1

1. Write a Java Program that uses both recursive and non-recursive functions to print the *nth* value of the Fibonacci sequence.
2. Write a Java Program that prints out all the prime numbers within a range.

Exercise -2

3. Write a Java Program that checks whether a given string is a palindrome or not.
4. Write a Java Program for sorting a given list of names in ascending order applying at least two

Sorting techniques (bubble, selection or insertion sort)

Exercise -3

5. Write a Java Program using String Tokenizer class, which reads a line of integers and then displays each integer and the sum of all integers
6. Write a Java Program to demonstrate sequence of Constructor calling in a Hierarchy. Implement parameterized constructors also for hierarchic calls.

Exercise -4

7. Write a Java Program check the compatibility for multiplication, if compatible multiply two matrices and find resultant matrix's transpose.

Exercise -5

8. Implement dynamic polymorphism, overloading (method and constructor) and overriding.
9. Write a java program how to access the super class variable and method and pass value.

Exercise -6

10. Write a Java Program that depicts file stream API to check whether a directory exists, number of files in a directory, file length and file content
11. Write a Java Program that implements a simple client/server application. The client sends data to a server. The server receives the data, uses it to produce a result and then sends the result back to the client. The client displays the result on the console. For ex: The data sent from the client is the radius of a circle and the result produced by the server is the area of the circle

Exercise -7

12. Write a Java Program that implements stack or queue with overflow and underflow conditions as Exceptions. Create your own exception classes for Overflow and Underflow

Exercise -8

13. Write a Java Program that converts infix notation to postfix.
14. Write a java program to evaluate postfix notation.

Exercise -9

15. Write a Java Program that creates 3 threads by extending Thread class. First thread displays "Good Morning" every 1 sec, the second thread displays "Hello" every 2 seconds and the third displays "Welcome" every 3 seconds.

    (Repeat the same by implementing Runnable)

16. Write a Java Program that correctly implements Producer-Consumer problem using the concept of Inter Thread Communication

Exercise -10

17. Write a Java Program that allows user to paint text, lines, rectangles and ovals (3d objects)
18. Write a Java Program that depicts mouse and key board action events

Exercise -11

19. Write a Java Program that works as a simple calculator. Use a grid layout to arrange buttons for the digits and for the + - x / % operations. Add a text field to display the result.

Exercise -12

20. Write a Java Program to Sort the DVD list based upon title, year of release, and Rating using comparator and Comparable Interface
21. Write a Java program to store list of Employee objects (Employee object will have empId, firstName, lastName, address, salary, deptId etc…) find out all employees for a given deptId and print the results to any output stream. [Use Lists / Iterators]

(if possible implement GUI for the above exercise)

II/IV B. TECH. FIRST SEMESTER
TECHNICAL ENGLISH (Required)

Course Code : CS 3L4 **Credits: 1**
Lab Hours: 2 periods/ week **Internal assessment:50Marks**
Tutorial:- **Semester end examination: --**

Prerequisites: English Language Communication Skills Lab

Course Objectives :

- To improve creativity in the students.
- To enable extemporaneous speaking, clarity of speech etc.
- To enhance the argumentative capacities.
- To Create a broad exposure to diverse thinking.
- To develop interpersonal and group skill.

Course Outcomes:

At the end of this course student will:

CO1) Acquire Expressionistic ability
CO2) Develop Assertiveness
CO3) Develop Leadership Qualities
CO4) Develop Communicate Competence
CO5) Refine Interview Skills

Syllabus:

UNIT-I

a. Just a minute on emerging technologies
b. Public speaking

UNIT- II

a. Debates on topics related to technology
b. Group discussion
c. Model Group Discussion
d. Dynamics

UNIT-III

a. Presentation skills
b. Power Point Presenations

UNIT-IV

a. Reporting
b. Meeting skills

UNIT-V

a. Interview Skills
b. Resume Preparation
c. Mock Interviews.

Reference Books:

1.Technical Communication – Ashraf Rizwi , McGraw Hill Publication
2. Business Communication and Personality Development – lesson for paradigm change in personality – Biswajit Das & Ipseeta Satpathy

II/IV B. TECH. SECOND SEMESTER
COMPILER DESIGN (Required)

Course Code : CS 4T1 **Credits: 3**
Lecture: 3 periods/week **Internal assessment: 30 Marks**
Tutorial: 1period/week **Semester end examination: 70 Marks**

Prerequisites: Formal Language & Automata Theory

Course Objectives:

1. An ability to use of formal attributed grammars for specifying the syntax and semantics of programming languages.
2. Working knowledge of the major phases of compilation, particularly lexical analysis, parsing, semantic analysis, and code generation Course
3. An ability to design and implement a significant portion of a compiler for a language chosen by the instructor.

Course Outcomes:
At the end of this course student will:

CO1) Understand about language processors and its phases.

CO2) Demonstrate about scanning of tokens and perform the syntax analysis by using parsing techniques

CO3) Perform Symantec analysis using attribute grammar and compare different memory management techniques in runtime environment

CO4) Ascertain optimization techniques for intermediate code forms and code generation

Syllabus:

UNIT 1
Overview of language processing: – preprocessors – compiler – assembler – Linkers & loaders, difference between compiler and interpreter- structure of a compiler –phases of a compiler.**Lexical Analysis**: - Role of Lexical Analysis – Input Buffering – Specification of Tokens – Recognition of Token – The Lexical Analyzer Generator Lex.

UNIT 2
Syntax Analysis: – Role of a parser – Context Free Grammar – Top Down Parsing – Recursive Descent Parsing — Non recursive Predictive Parsing- FIRST and FOLLOW –

LL(1) Grammar – Error Recovery in Predictive Parsing.

UNIT 3

Bottom up Parsing: – Reductions – Handle Pruning - Shift Reduce Parsing - Introduction to simple LR – Why LR Parsers – Model of an LR Parsers — Construction of SLR Tables.

More powerful LR parsers: - Construction of CLR (1) - LALR Parsing tables.

UNIT 4

Runtime Environment: - Storage organization - Stack allocation – Static allocation - Heap management - Parameter passing mechanisms.

Intermediate code: - DAG - Three address code – Quadruples - Triples - Indirect Triples.

UNIT 5

Basic Blocks: – DAG representation of Block. Machine independent code optimization - Common sub expression elimination - Constant folding - Copy propagation -Dead code elimination - Strength reduction - Loop optimization.

Machine dependent code optimization: - Peephole optimization – Register allocation - Instruction scheduling - Inter Procedural Optimization - Garbage collection via reference counting.

Learning Resource

Text Books

1. Compilers: Principles, Techniques and Tools: 2nd Edition, Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ulman; 2nd Edition, Pearson Education.

2. Modern Compiler Implementation in C- Andrew N. Appel, Cambridge University Press

References

1. lex &yacc – John R. Levine, Tony Mason, Doug Brown, O'reilly
2. Modern Compiler Design- Dick Grune, Henry E. Bal, Cariel T. H. Jacobs, Wiley reamtech.
3. Engineering a Compiler-Cooper & Linda, Elsevier.
4. Compiler Construction, Louden, Thomson.
5. Principles of compiler design, V. Raghavan, 2nd ed, TMH, 2011.
6. http://www.nptel.iitm.ac.in/downloads/106108052/

II/IV B. TECH. SECOND SEMESTER
DESIGN AND ANALYSIS OF ALGORITHMS (Required)

Course Code : CS 4T2 **Credits: 3**
Lecture: 3 periods/ week **Internal assessment: 30 Marks**
Tutorial: 1period/week **Semester end examination: 70 Marks**

Prerequisites: Program Design

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

1. Analyze the asymptotic performance of algorithms.
2. Ability to analyze asymptotic runtime complexity of algorithms including formulating recurrence relations.
3. Ability to understand and design algorithms using greedy strategy, divide and conquer approach, dynamic programming, Demonstrate a familiarity with major algorithms and data structures.

Course Outcomes:

At the end of this course student will:

CO1) Understand the basic notation for analyzing the performance of the algorithms.

CO2) Use divide-and-conquer techniques for solving suitable problems

CO3) Use greedy approach to solve an appropriate problem for optimal solution.

CO4) Apply dynamic programming approach to solve suitable problems

CO5) Understand the limitations of algorithm power and study how to cope with the limitations of algorithm power for various problems

Syllabus:

UNIT 1
Introduction: Notion of an Algorithm–Fundamentals of Algorithmic Problem Solving – Fundamentals of the Analysis of Algorithm Efficiency – Analysis Framework –Asymptotic Notations and Basic efficiency classes.

UNIT 2
Divide and Conquer Method: General Method, Applications: Binary search, Quick sort, Merge sort and Analysis of divide and conquer runtime recurrence relations.

UNIT 3

Greedy Method: General method, Applications: Minimum cost spanning tree (prim's and kruskal's algorithm), Dijkstra's alorithm.

UNIT 4

Dynamic programming: General Method, Applications: Floyd's algorithm, Optimal Binary Search Tree, 0/1 knapsack problem

UNIT 5

Back tracking: General Method, Applications: Sum of Subsets, Hamiltonian Cycles.
Branch and bound: The Method – Assignment problem, Travelling Salesman Problem **-** Introduction to NP-Hard and NP-Complete Problems.

Learning Resource

Text Books

1. Introduction to the Design & Analysis of Algorithms, Anany Levitin, 2nd Edition, Pearson Education 2007.

References

1. "Introduction to Algorithms", 3rd Ed., T. H. Cormen, C. E. Leiserson, R. L. Rivest, Clifford Stein, PHI.
2. "Computer Algorithms", Ellis Horowitz and Sartaj Sahni, Silicon press, 2008.

II/IV B. TECH. SECOND SEMESTER

FILE STRUCTURES

(Required)

Course Code : CS 4T3 **Credits: 3**

Lecture: 3 periods/ week **Internal assessment: 30 Marks**

Tutorial: 1period/week **Semester end examination: 70 Marks**

Prerequisites: Data Structures

Course Objectives:

1. Provide an introduction to the fundamental file operations and storage systems.
2. Introducing fundamental concepts of file structure.
3. Introducing the most important high-level file structures tools which include indexing, co sequential processing, B trees, Hashing.
4. Applying the techniques in the design of C++ programs for solving various file management problems.

Course Outcomes:

At the end of this course student will:

CO1) Understand the fundamental concepts of file processing operations and storage structures

CO2) Apply object orientation concepts to manipulate records

CO3) Apply concepts of sorting and merging on multiple files

CO4) Analyze the sequential and indexing file accessing techniques with appropriate data structures

CO5) Illustrate the usage of hashing techniques to organize file structures

Syllabus:

UNIT 1

Fundamental File Processing Operations : Physical Files and Logical Files, Opening Files, Closing Files, Reading and Writing, Seeking, Special Characters in Files, The Unix Directory Structure, Physical Devices and Logical Files.

Secondary Storage and System Software: Disks, Magnetic Tape, Disk versus Tape,

Introduction to CD-ROM, Physical Organization of CD-ROM, CD-ROM Strengths and Weaknesses; Storage as a Hierarchy, A Journey of a Byte, Buffer Management, I/O in UNIX.

UNIT 2

F**undamental File structure Concepts**: Field and Record Organization, Using Classes to Manipulate Buffers, Using Inheritance for Record Buffer Classes, Managing Fixed Length and Fixed Field Buffers.

Managing Files of Records: Record Access, More about Record Structures, Encapsulating Record I/O Operations in a Single Class, File Access and File Organization.

Co-sequential Processing: An Object-Oriented Model for Implementing Cosequential Processes.

UNIT 3

Indexing: What is an Index?, A Simple Index for Entry -Sequenced Files. **Multilevel Indexing and B-Trees**: Introduction: The Invention of B-Tree, Statement of the Problem, Indexing with Binary Search Trees, Multilevel Indexing, a Better Approach to Tree Indexes, B-trees: Working up from the Bottom, Example of Creating a B-Tree, B-Tree Methods Search, Insert and Others.

UNIT 4

Indexed Sequential File Access and Prefix B+ Trees: Indexed Sequential Access, Maintaining a Sequence Set, Adding a Simple Index to the Sequence Set, The Content of the Index: Separators Instead of Keys, The Simple Prefix B+ Tree, Simple Prefix B+ Tree Maintenance.

UNIT 5

Hashing: Introduction, A Simple Hashing Algorithm, Hashing Functions and Record Distributions, How Much Extra Memory Should Be Used?, Collision Resolution by Progressive Overflow, Storing More Than One Record per Address: Buckets, Making Deletions, Other Collision Resolution Techniques, Patterns of Record Access.

Learning Resource

Text Books

1. File Structures: An Object-Oriented Approach with C++, Michael J. Folk, Greg Riccardi, Bill Zoellick, Third Edition, Pearson Education.

References

1. Data Management and File Structures, Mary E.S. Loomis, Second Edition, PHI.
2. File Organization and Processing, Alan L. Tharp, Wiley India Edition.

II/IV B. TECH. SECOND SEMESTER
PRINCIPLES OF PROGRAMMING LANGUAGES(Required)

Course Code : CS4T4 **Credits: 3**
Lecture: 3 periods/ week **Internal assessment: 30 Marks**
Tutorial: 1period/week **Semester end examination: 70 Marks**

Prerequisites: Program Design

Course Objectives:

1. Increased capacity to express ideas.
2. Improved knowledge in choosing appropriate languages.
3. Increased ability to learn new languages.
4. Understand the differences between different programming paradigms.
5. Increased capacity to develop programs in different programming languages.

Course Outcomes:

At the end of this course student will:

CO1) Understand the architecture and implementation methods of Programming languages

CO2) Design parse trees for syntax and semantics of programming languages.

CO3) Understand characteristics and features of various data types and control structures in programming languages

CO4) Illustrate the modularity in programs with different parameter passing techniques

CO5) Describe the features of logical & functional programming languages

Syllabus:

UNIT 1

Preliminary Concepts: - Reasons for studying, concepts of programming languages, Programming domains, Language Evaluation Criteria, influences on Language design, Language categories, Programming Language Implementation – Compilation and Virtual Machines, programming environments.

UNIT 2

Syntax and Semantics: - general Problem of describing Syntax and Semantics, formal methods of describing syntax - BNF, EBNF for common programming languages features, parse trees, ambiguous grammars, attribute grammars, dynamic semantics

UNIT 3

Data types: - Introduction, primitive, character, user defined, array, associative, record, union, pointer and reference types. Names, Variable, concept of binding, type checking, strong typing, type compatibility, named constants, variable initialization.

UNIT 4

Expressions and Statements:- Arithmetic relational and Boolean expressions, Short circuit evaluation mixed mode assignment, Assignment Statements, Control Structures – Statement Level, Compound Statements, Selection, Iteration, Unconditional Statements, and guarded commands.

UNIT 5

Subprograms and Blocks:- Fundamentals of sub-programs, Scope and lifetime of variable, static and dynamic scope, Design issues of subprograms and operations, local referencing environments, parameter passing methods, overloaded sub-programs, generic sub-programs, parameters that are sub-program names, co routines.

Functional Programming Languages: - Introduction, fundamentals of FPL, LISP, ML, Haskell, application of Functional Programming Languages and comparison of functional and imperative Languages.

Learning Resource

Text Books

1. Concepts of Programming Languages, Robert .W. Sebesta 6/e, Pearson Education.

2. Programming Languages –Louden, Second Edition, Thomson

References

1. Programming languages –Ghezzi, 3/e, John Wiley
2. Programming Languages Design and Implementation – Pratt and Zelkowitz, Fourth Edition PHI/Pearson Education
3. Programming languages –Watt, Wiley Dreamtech

II/IV B. TECH. SECOND SEMESTER
COMPUTER ORGANIZATION(Required)

Course Code : CS4T5 **Credits: 3**
Lecture: 3 periods/ week **Internal assessment: 30 Marks**
Tutorial: 1period/week **Semester end examination: 70 Marks**

Prerequisites: Digital Logic Design

Course Objectives:

Ability to:

1. Design simple combinational circuits.
2. Design basic building blocks of a computer like ALU, registers, processor and memory at gate level.
3. Analyze the organization of various memory types and I/O devices.
4. Understand the concept of Data Transfer in various components of a computer system.

Course Outcomes:
At the end of this course student will:

CO1) Design simple combinational circuits
CO2) Design basic building blocks of a computer like ALU, registers, processor and memory at gate level.
CO3) Analyze the organization of various memory types and I/O devices.
CO4) Understand the concept of data transfer in various components of a computer system

Syllabus:

UNIT 1

DIGITAL LOGIC CIRCUITS: Digital Computers, Logic Gates, Boolean Algebra: Compliment of a Function, Map Simplification: Product-of-Sums Simplification, Don't-Care Conditions, Combinational Circuits: Half-Adder, Full-Adder, Flip-Flops: SR Flip-Flop, D Flip-Flop, JK Flip-Flop, T Flip-Flop, Edge-Triggered Flip-Flops, Excitation Tables.

UNIT 2

CENTRAL PROCESSING UNIT: General register Organization, Stack Organization: Register Stack, Memory Stack, Reverse Polish Notation, Instruction Formats: Three-Address Instructions, Two-Address Instructions, One-Address Instructions, Zero-Address Instructions ,RISC Instructions, Addressing Modes, Data Transfer and Manipulation, Program Control, Reduced Instruction Set Computer (RISC): CISC Characteristics, RISC Characteristics.

UNIT 3

INPUT-OUTPUT ORGANIZATION: Peripheral Devices: ASCII Alphanumeric Characters, Input-output Interface: I/O Bus and Interface Modules, Asynchronous Data Transfer: Strobe Control, Handshaking, Asynchronous Communication Interface, First-In, First-Out Buffer, Modes of Transfer: Interrupt-Initiated I/O, Priority Interrupt: Daisy-Chaining Priority, Parallel Priority Interrupt, Priority Encoder, Direct Memory Access (DMA): DMA Controller.

UNIT 4

MEMORY ORGANIZATION: Memory Hierarchy, Main Memory: RAM and ROM Chips, Memory Address Map, Memory Connection to CPU, Auxiliary memory: Magnetic Disks, Magnetic Tapes, Associative Memory: Hardware Organization, Match Logic, Read & Write Operations, Cache Memory: Associative Mapping, Direct Mapping, Set-Associative Mapping, Virtual Memory: Address Space and Memory Space, Address Mapping using Pages, Associative Memory Page Table, Page Replacement.

UNIT 5

MULTIPROCESSORS: Characteristics of multiprocessors, Interconnection structures, Inter processor arbitration,Interprocessor communication and synchronization.

Learning Resource

Text Books

1. Computer System Architecture, Morris M. Mano, 3rd edition,Pearson/Prentice Hall India.

References

1. Computer Organization and Achitecture, William Stallings,8th edition,PHI
2. Computer Organization, Carl Hamachar, Vranesic,Zaky, 5th edition, McGraw Hill.

II/IV B. TECH. SECOND SEMESTER
COMPILER DESIGN LAB (Required)

Course Code : CS 4L1 **Credits: 2**
Lab Hours: 3 periods/ week **Internal assessment: 25 Marks**
Tutorial:- **Semester end examination: 50 Marks**

Prerequisites: Compiler Design

Course Objectives:

1. To implement Lexical Analyzer using Lex tool & Syntax Analyzer or parser using YACC Tool
2. To implement NFA and DFA from a given regular expression
3. To implement front end of the compiler by means of generating Intermediate codes.
4. To implement code optimization techniques.

Course Outcomes:

At the end of this course student will:

CO 1) Design Lexical analyzer for given language using C and LEX tools.
CO 2) Design and convert BNF rules into YACC form to generate various parsers.
CO 3) Generate machine code from the intermediate code forms
CO 4) Implement Symbol table

Syllabus:

1. Design a Lexical analyzer for the given language. The lexical analyzer should ignore redundant spaces, tabs and new lines. It should also ignore comments. Although the syntax specification states that identifiers can be arbitrarily long, you may restrict the length to some reasonable value.
2. Implement the lexical analyzer using JLex, flex or lex or other lexical analyzer generating stools.
3. Design Predictive parser for the given language.
4. Design LALR bottom up parser for the given language.
5. Convert the BNF rules into Yacc form and write code to generate abstract syntax tree.
6. Write program to generate machine code from the abstract syntax tree generated by the parser.

7. Implementation of Symbol Table.

8. Generation of Code for a given Intermediate Code.

Learning Resource

Text Books

1. Compilers: Principles, Techniques and Tools: 2nd Edition, Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ulman; 2nd Edition ,Pearson Education.

2. Modern Compiler Implementation in C- Andrew N. Appel, Cambridge University Press.

Reference Books

1. lex &yacc – John R. Levine, Tony Mason, Doug Brown, O'reilly

2. Modern Compiler Design- Dick Grune, Henry E. Bal, Cariel T. H. Jacobs, Wiley dreamtech.

3. Engineering a Compiler-Cooper & Linda, Elsevier.

4.Compiler Construction, Louden, Thomson.

5. Principles of compiler design, V. Raghavan, 2nd ed, TMH, 2011.

6.http://www.nptel.iitm.ac.in/downloads/106108052/

II/IV B. TECH. SECOND SEMESTER
FILE STRUCTURES LAB(Required)

Course Code : CS 4L2 **Credits: 2**
Lab Hours: 3 periods/ week **Internal assessment: 25 Marks**
Tutorial:- **Semester end examination: 50 Marks**

Prerequisites: File Structures

Course Objectives:

1. Provide a solid introduction to the topic of file structure design.
2. Discuss, in detail, the data structures necessary for achieving its efficiency objectives.
3. Introducing the most important high-level file structures tools which include indexing, co sequential processing, B trees, Hashing.

Course Outcomes:
At the end of this course student will:

CO1) Implement various operations on files

CO2) Apply indexing techniques on files

CO3) Employ multiple lists merging concept for files.

CO4) Synthesize and implement the multilevel indexing concept(B Trees) on files

CO5) Apply the hashing technique to resolve collision of records.

Syllabus:

1. Implement the following programs using C++ language.
    a. Write a program to create a class Student. Each student object represents information about a single student. Members should be included for identifier, name, address, date of first enrollment, and number of credit hour completed. Methods should be included for initialization (constructors), assignment (overloaded '=' operator), and modifying field values, including a method to increment the number of credit hours.

    b. Write a program to create a class **CourseRegistration**. Each object represents the enrollment of a student in a course. Members should be included for a course identifier, student identifier, number of credits hours, and course grade. Method should be included as appropriate.

    c. Create a list of student and course registration information. This information will be used in subsequent exercises to test and evaluate the capabilities of the programming project.

2. Write a C++ program to read and write student objects and courseregistration objects with fixed-length records and the fields delimited by "|". Implement pack ( ), unpack ( ), modify ( ) and search ( ) methods.
3. Write a C++ program to read and write student objects and courseregistration objects with Variable - Length records using any suitable record structure. Implement pack ( ), unpack ( ), modify ( ) and search ( ) methods.
4. Write a C++ program to write student objects and courseregistration objects with Variable - Length records using any suitable record structure and to read from this file a student record using RRN.
5. Write a C++ program to implement simple index on primary key for a file of student objects and courseregistration objects. Implement add ( ), search ( ), delete ( ) using the index.
6. Write a C++ program to implement index on secondary key, the name, for a file of student objects and courseregistration objects . Implement add ( ), search ( ), delete ( ) using the secondary index.
7. Write a C++ program to read two lists of names and then match the names in the two lists using Cosequential Match based on a single loop. Output the names common to both the lists.
8. Write a C++ program to read k Lists of names and merge them using k-way merge algorithm with k = 8.
9. Write a C++ program to implement B-Tree for a given set of integers and its operations insert ( ) and search ( ). Display the tree.
10. Use class B+ Tree to create a B-tree index of a student record file with student identifier as key. Write a driver program to create a B-tree file from an existing student record file. Display the tree.
11. Write a C++ program to store and retrieve student data from file using hashing. Use any collision resolution technique.
12. Write a C++ program to reclaim the free space resulting from the deletion of records using linked lists.

Learning Resources:

1. File Structures: An Object-Oriented Approach with C++, Michael J. Folk, Greg Riccardi, Bill Zoellick, Third Edition, Pearson Education.
2. Data Management and File Structures, Mary E.S. Loomis, Second Edition, PHI.
3. File Organization and Processing, Alan L. Tharp, Wiley India Edition.

II/IV B. TECH. SECOND SEMESTER
COMPUTER ORGANIZATION LAB(Required)

Course Code : CS 4L3 **Credits: 2**
Lab Hours: 3 periods/ week **Internal assessment: 25 Marks**
Tutorial:- **Semester end examination: 50 Marks**

Prerequisites: Computer Organization

Course Objectives:

1. Understanding the behavior of Logic Gates, Adders, Decoders, Multiplexers and Flip-Flops.
2. Understanding the behavior of ALU, RAM, STACK and PROCESSOR from working modules and the modules designed by the student as part of the experiment.

At the end of this course student will:

CO1) Analyze the behaviour of logic gates
CO2) Design combinational circuits for basic components of computer system and applications.
CO3) Analyze the operational behaviour and applications of various flip-flop
CO4) Design Arithmetic logic units and different types of memory blocks.

Syllabus:

1. Introduction to Verilog HDL/VHDL
2. Verify the behavior of logic gates using truth tables (AND, OR, NOT, XOR, NAND, NOR)
3. Implementing HALF ADDER, FULL ADDER using basic logic gates
4. Implementing Binary -to -Gray, Gray -to -Binary code conversions
5. Implementing 3-8 line DECODER.
6. Implementing 4x1 and 8x1 MULTIPLEXERS.
7. Verify the excitation tables of various FLIP-FLOPS
8. Design of an 8-bit Input/Output system with four 8-bit Internal Registers.
9. Design of an 8-bit ARITHMETIC LOGIC UNIT.

. Design of 24x8 (16 byte) RAM.

. Design of 24x8 (16 byte) STACK.

. Implementation of a 4-bit PROCESSOR.

Learning Resources:

References:

1)A Verilog HDL Primer by J. Bhasker Bk&Hardcover; Published by Star Galaxy Press. ISBN: 0-9656277-4-8

2)Verilog HDL : A Guide to Digital Design and Synthesis by Samir Palnitkar Published by Prentice Hall Publication date: March 1996

II/IV B. TECH. SECOND SEMESTER
PERSONALITY DEVELOPMENT COURSE(Required)

Course Code : CS 4L4 **Credits: 1**
Lab Hours: 2 periods/ week **Internal assessment:50 Marks**
Tutorial:- **Semester end examination: --**

Prerequisites: English Language Communication Skills lab, Technical English

Objectives:

To introduce fundamentals of various aspects of personality traits. To give them adequate exposure to the basic aspects which mould the personality To enable them to develop humble nature. To create in them the love for human values.

Outcomes:

Students will be able to

CO1) Improve Leadership skills.
CO2) Acquire Non verbal skills.
CO3) Develop Team culture.
CO4) Improve Managerial qualities and communication skills.
CO5) Refine the Body Language.

Syllabus:

UNIT-I

PERSONALITY:

1. Grooming one‟s personality
2. Influence of heredity and environment on personality
3. Different personality types.

UNIT-II

PERSONALITY DEVELOPMENT

1. Freudian Analysis
2. Vivekananda concept

UNIT-III

LEADERSHIP QUALITIES

1. Communication skills
2. Attitude
3. Empathy
4. Adaptability
5. Conflict Resolution

UNIT-IV

SOFT SKILLS IN WORKPLACE

1. Time management
2. Planning &organization
3. Parkinson‟s law

5 Team work

6. Assertiveness

UNIT-V

BODY LANGUAGE

1. Aggressiveness
2. Submissiveness
3. Attentiveness
4. Nervousness
5. Defensiveness
6. Handshake

Text Book:

1. Personality development &soft skills BarunK.Mith Oxford

Reference Books:

1.Personal &emotional competence, V.Bhaskara Rao, B.S.P

2.Step by step –Niruparani.K, JayasreeMohanra, Pearson

III/IV B.Tech. FIRST SEMESTER

DATABASE MANAGEMENT SYSTEMS

Course Code : CS5T1	**Credits: 3**
Lecture: 3 periods/ week	**Internal assessment: 30 Marks**
Tutorial: 1period/week	**Semester end examination: 70 Marks**

Prerequisites: Data Structures, File Structures

Course Objectives:

1. The main objective of this course is to enable students to the fundamental concepts of database analysis and design.
2. To recognize the importance of database analysis and design in the implementation of any Database application and to understand the process of drawing the ER-Diagrams.
3. It also gives the knowledge of the roles of transaction processing and concurrency control.

Course Outcomes:

CO1) Understand the basic principles of database management systems.

CO2) Design Entity-Relationship diagrams to represent simple database application scenarios.

CO3) Develop relational tables and sql queries for a given context in relational database.

CO4)Apply normalization techniques to a given database application.

CO5) Describe transaction processing and concurrency control.

Syllabus:

UNIT 1

Introduction to Databases: Characteristics of the Database Approach, Advantages of using the DBMS Approach, A Brief History of Database Applications.

Overview of Database Languages and Architectures: Data Models, Schemas and Instances, Three-Schema Architecture and Data Independence, Database Languages and Interfaces, Database System environment, Centralized and Client-Server Architecture for DBMSs.

UNIT 2

Relational Model: The Relational Model Concepts , Relational Model Constraints and Relational Database Schemas.**SQL:** Data Definition, Constraints, and Basic Queries and

Updates,**SQL: Advanced Queries**, Assertions, Triggers, and Views.

Formal Relational Languages: Relational Algebra: Unary Relational Operations: Select and Project, Relational Algebra Operations from Set Theory, Binary Relational Operations: Join and Division, Examples of Queries in Relational Algebra.

UNIT 3

Conceptual Data Modeling : High-Level Conceptual Data Models for Database Design, A Sample Database Application, Entity Types, Entity Sets, Attributes and Keys, Relationship Types, Relationship Sets, Roles, and Structural Constraints, Weak Entity Types, Refining the ER Design, ER Diagrams, Naming Conventions and Design Issues, Relationship Types of Degree Higher Than Two Relational Database Design Using ER-to-Relational Mapping**.**

UNIT 4

Database Design Theory : Functional Dependencies, Normal forms based on Primary Keys, Second and Third Normal Forms, Boyce-Codd Normal Form, Multi valued Dependencies and Fourth Normal Form, Join Dependencies and Fifth Normal Form.

UNIT 5

Transaction Processing: Introduction, Transaction and System Concepts, Desirable Properties of Transactions, Characterizing Schedules Based on Recoverability & Serializability, Transaction Support in SQL.

Introduction to Concurrency Control: Two-Phase Locking Techniques: Types of Locks and System Lock Tables, Guaranteeing Serializability by Two-Phase Locking.

Introduction to Recovery Protocols – Recovery Concepts, No-UNDO/REDO Recovery Based on Deferred Update, Recovery Techniques Based on Immediate Update, Shadow Paging.

Learning Resource

Text Books

1. DATABASE SYSTEMS Models, Languages, Design and Application Programming, 6th Edition, Ramez Elmasri ,Shamkant B.Navathe , Pearson.

References

1. Data base System Concepts, 5th Edition, Abraham Silberschatz, Henry F Korth, S.Sudarshan,Mc Graw Hill.

2. Data base Management Systems, 3rd Edition, Raghurama Krishnan, Johannes Gehrke, TMH.

3. Introduction to Database Systems, 8th Edition , C.J.Date, Pearson

III/IV B. TECH. FIRST SEMESTER
MICROPROCESSOR AND INTERFACING (Required)

Course Code : CS 5T2 **Credits: 3**
Lecture: 3 periods/ wee k **Internal assessment: 30 Marks**
Tutorial: 1period/week **Semester end examination: 70 Marks**

Prerequisites: Computer Organization and Logic Design

Course Objectives:
At the end of the course, students are expected to have:

1. Ability to design and conduct experiments related to microprocessor based
2. system design and to analyze their outcomes.
3. learn how the hardware and software components of a microprocessor-based system work together to implement system-level features;
4. learn both hardware and software aspects of integrating digital devices (such as memory and I/O interfaces) into microprocessor-based systems;
5. get practical experience in applied digital logic design and assembly-language programming; and
6. be exposed to the tools and techniques used by practicing engineers to design, implement, and debug microprocessor-based systems (during the Lab).

Course Outcomes:
At the end of this course student will:

CO1) Identify the basic elements and functions of microprocessor

CO2) Describe the architecture of microprocessor and its peripheral devices

CO3) Demonstrate fundamental understanding on the operation between the microprocessor and its interfacing devices

CO4) Understand the evolution of processor architectures

Syllabus:

UNIT 1

Introduction to Microprocessors, Internal Architecture of 8086, Addressing modes of 8086.

UNIT 2

8086 Assembly Language Programs: 8086 instruction set, Assembler directive, program development method, Writing simple 8086 programs for use with an assembler.

UNIT 3

8086 Interrupts: 8086 Interrupts and Interrupt responses, hardware interrupt application. Interfacing: Digital interfacing, Programming parallel port and handshake I/O, Interfacing a Microprocessor to keyboards & displays

UNIT 4

Introduction to 80286, 80386, 80486 microprocessor, Single chip microcontrollers.

UNIT 5

Introduction to Pentium Processor architecture, Introduction and Evolution of Multicore processors, dual Core and Core Duo Basic characteristics,
Architecture and comparison with other CPU's.

Learning Resource

Text Books

1. Micro Computer System 8086/8088 Family Architecture, Programming and Design - By Liu and GA Gibson, PHI, 2nd Ed.

References

1. Microprocessor Architecture, Programming, and Applications With the 8085 , Ramesh S Gaonkar,prentice hall 5e.
2. The X86 Microprocessors, architecture, Programming and Interfacing(8086 to Pentium), Lyla B Das, Pearson.
3. A.P. Mathur ", Introduction Microprocessor–IIIrd Edition", (TMH)
4. Tabak. D," Advanced Microprocessor-2nd edition," (TMH)
5. The Intel Microprocessors by Barry B.Brey
6. The 8086 Microprocessor: Programming & Interfacing the PC, Ayala: Cengage

III/IV B. TECH. FIRST SEMESTER
COMPUTER NETWORKS
(Required)

Course Code : CS 5T3 **Credits: 3**

Lecture: 3 periods/ week **Internal assessment: 30 Marks**

Tutorial: 1period/week **Semester end examination: 70 Marks**

Prerequisites: Fundamental Operating System Concepts , Introduction to C Programming and Data Structures

Course Objectives:

- Familiarize the student with the basic taxonomy and terminology of the computer networking area.
- Introduce the students to basic principles of networking using the goals like protocol layering and top down approach.
- Build an understanding of the basics of the internetworking and routing used in the computer networks.
- To provide guidelines in developing network applications

Course Outcomes:

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

CO1) Independently understand basic computer network technology

CO2) Identify the different types of network topologies.

CO3) Enumerate the layers of TCP/IP. Explain the functions of each layer

CO4) Familiarity with the protocols of computer networks and routing mechanisms

CO5) Classify different types of physical layer transmissions and various transmission media.

Syllabus:

UNIT 1

Introduction: Overview of the Internet, Networks, LAN , WAN ,point to point WAN, Switched WAN, Switching, Switched network, circuit switched network, packet switched network,

Protocol Layering: TCP/IP protocol suite, Layers in TCP/IP, The OSI Model, Description of each layer, OSI reference model definition, comparison of TCP/IP and OSI model.

UNIT 2

Introduction: Providing Services , Standard Application layer protocols, Application layer paradigms, Overview of Client server paradigm and Peer to peer networks.

Standard Client Server Applications like WWW its architecture, url HTTP – no persistent versus persistent connections, message formats FTP-communication

over data connection, architecture, file transfer examples , sending & receiving mail and

Overview of SMTP, Telnet , NVT,SSH ,DNS

UNIT 3

Introduction to transport layer ,Transport layer protocols, User Datagram Protocol, Transmission control protocol:

UNIT 4

Network layer: Introduction ,Network Layer Services, Packet Switching , Network Layer Performance, Network Layer Congestion.

Network layer protocols-IPv4 Datagram Format, IPV4 Addresses notation , Classful Addressing, Classless Addressing, Subnetting, NAT, ICMPv4. IPv6,Packet Format , Addressing, Translation from IPV4 to IPV6

Routing algorithms like DV, Link state, dijkstra's spanning trees.

UNIT 5

Data Link Layer: Introduction to DLC, Nodes and Links, Types of Links.

DLC: Framing, Flow and Error Control, Error Detection and Correction :Types of Errors
Coding :Block coding , linear coding , cyclic coding.

Overview of Two DLC Protocols: HDLC , PPP.

MAC protocols: Aloha, CSMA , CSMA / CD, CSMA/CA, controlled access,
Wired LANS-Ethernet protocol
Physical Layer: Transmission media, Guided media :twisted pair cable, coaxial cable, fibre optic cable and Unguided media

Learning Resource

Text Books

1. Computer Networks: A Top –Down Approach, Behrouz A. Forouzan and Firouz, Mosharraf,2012 , Tata McGraw Hill.

References

1. Computer Networking: A Top down Approach Featuring the Internet, Kurose & Rose, 3rd Edition, Pearson.
2. Computer Networks A Systems Approach,5/e, Larry L. Peterson and Bruce S. Davie, Morgan Kaufmann(Elsevier).
3. Data and Computer Communication, Eighth Edition, William Stallings, Pearson.

III/IV B. TECH. FIRST SEMESTER
SOFT COMPUTING(Required)

Course Code : CS 5T4 **Credits: 3**
Lecture: 3 periods/ week **Internal assessment: 30 Marks**
Tutorial: 1period/week **Semester end examination: 70 Marks**

Prerequisites: NIL

Course Objectives:

1. Soft computing refers to principle components like fuzzy logic, neural networks and genetic algorithm, which have their roots in Artificial Intelligence.
2. Healthy integration of all these techniques has resulted in extending the capabilities of the technologies to more effective and efficient problem solving methodologies

Course Outcomes:

At the end of this course student will:

CO1) Demonstrate Fuzzy set theory

CO2) Interpret fuzzy systems

CO3) Apply ANN Back propagation algorithm for classification

CO4) Apply ANN training algorithms for solving real world problems

CO5) Explain fundamentals and operators of Genetic Algorithm.

Syllabus:

UNIT 1

Fuzzy Set Theory: Fuzzy Versus Crisp, Crisp Sets, Fuzzy Sets, Crisp Relations, Fuzzy Relations

UNIT 2

Fuzzy Systems: Crisp Logic, Predicate Logic, Fuzzy Logic, Fuzzy Rule Based Systems, Defuzzification Methods, Applications.

UNIT 3

Fundamentals of Neural Networks: Basic Concepts of Neural Networks, Human Brain, Model of an Artificial Neuron, Neural Network Architectures, Characteristics of Neural Networks, Learning Methods, Taxonomy of Neural Network Architectures. Back Propagation Networks: Architecture of a Back Propagation Network, Back Propagation Learning, Illustration, Applications (Classification of Soil only).

UNIT 4

Associative Memory: Autocorrelations, Heterocorrelators, Associative Memory for Real-Coded Pattern Pairs, Applications (Recognition of Characters only).

Adaptive Resonance Theory: Introduction, ART1, ART2, Applications (Recognition of Characters only), Sensitives of Ordering of Data.

UNIT 5

Fundamentals of Genetic Algorithms: Genetic Algorithms: History, Basic Concepts, Creation of Offsprings, Working Principle, Encoding, Fitness Function, Reproduction.

Genetic Modeling: Inheritance Operators, Cross Over, Inversion, And Deletion, Mutation Operator, Bit-Wise Operators, Bit-Wise Operators used in GA, Generational Cycle, Convergence of Genetic Algorithms, Hybrid Systems(10.1), NN & FL & GA Hybrids(10.2)

Learning Resource

Text Books

1. S.Rajasekaran, G.A. Vijayalakshmi Pai, Neural Networks, fuzzy logic, and genetic algorithms - Genetic Algorithm, PHI Learning Private Limited- 2010.
2. S.N.Sivanandam, S.N.Deepa Wiley India , Principles of SOFT COMPUTING, Second Edition 2011.

References

1. Timothy J. Ross, “Fuzzy Logic with Engineering Applications” Wiley India.
2. Siman Haykin,”Neural Netowrks”Prentice Hall of India.
3. Kumar Satish, “Neural Networks” Tata Mc Graw Hill

III/IV B. TECH. FIRST SEMESTER
OPERATING SYSTEMS(Required)

Course Code : CS 5T5 **Credits: 3**
Lecture: 3 periods/ week **Internal assessment: 30 Marks**
Tutorial: 1period/week **Semester end examination: 70 Marks**

Prerequisites: Data Structures

Course Outcomes:

At the end of this course student will:

CO1) Understand the structure and functionalities of Operating System

CO2) Apply CPU scheduling algorithms, deadlock prevention and detection algorithms and different page replacement algorithms

CO3) Illustrate different problems and solutions related to process synchronization

CO4) Describe the concepts of paging and segmentation for memory management

CO5) Analyze the operating system support for virtual memory, disk management

Syllabus:

UNIT-I:

Computer System and Operating System Overview: Operating System Objectives & Functions, Computer System Organization & Architecture, Operating System Structure & Operations. System Structure: OS Services, System Calls, Types of system calls.

UNIT-II:

Process Management: Process Concept, Process scheduling, Operations on processes, Co-Operating Processes, Interprocess Communication.
Threads: Overview, Multithreading Models, user and kernel threads.
Process Scheduling: Basic Concepts, Scheduling Criteria, Scheduling Algorithms (FCFS, SJF, Priority, RR).

UNIT-III:

Process Synchronization: Critical Section Problem, Peterson's Solution Synchronization Hardware, Semaphores, Classical problems of synchronization, Monitors.
Deadlocks: System Model, Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance (including Banker's Algorithm), Deadlock Detection & Recovery

UNIT-IV:

Memory Management: Logical vs. physical address space, Swapping, Contiguous Memory Allocation, Paging, Structure of the Page Table, Segmentation.

Virtual Memory Management: Page fault, Demand Paging, Performance, Page Replacement & its Algorithms (FIFO, LRU Optimal, Clock), Allocation of frames, Thrashing

UNIT-V:

File System: File Concept, Access Methods, Directory & Disk Structure, File System Structure, Directory Implementation (linear list, hash table), and Allocation methods (contiguous, linked, and indexed).

Disk Management: Overview of Mass Storage Structure, Disk Scheduling (FCFS, SSTF, SCAN, C-SCAN)

Learning Resource

Text Books

1 Operating System Concepts- Abraham Silberchatz, Peter B. Galvin, Greg Gagne,8thEdition, John Wiley.

References

1. Operating Systems' – Internal and Design Principles -- Stallings, Sixth Edition,Pearson education.
2. Operating System Design & Implementation -- Tanenbaum A.S. -- PHI
3. Operating Systems -- Stalling, William -- Maxwell McMillan International Editions.
4. An Introduction to Operating Systems -- Dietel H. N., -- Addison Wesley.
5. Advanced programming in the UNIX environment -- W.Richard Stevens -- pearson Education
6. UNIX and Shell Programming -- **Behrouz A. Forouzan**, **Richard F. Gilberg.**

III/IV B.TECH. FIRST SEMESTER
DATA BASE MANAGEMENT SYSTEMS LAB (Required)

Course Code : CS5L1 **Credits: 2**
Lab Hours : 3 periods/week **Internal assessment: 25 marks**
Tutorial : -- **Semester end examination: 50 marks**

Prerequisite: Data base Management Systems

Course Outcomes:

CO1) Design and implement a database schema for a given problem-domain

CO2) Formulate query for a database using DDL/DML commands

CO3) Apply integrity constraints on a database

CO4) Develop programs including procedures, stored functions, cursors and packages for data manipulation.

The Following Experiments are to be conducted using MySQL Database.

Syllabus:

EXP-1

1. Introduction to MySQL Workbench.
2. How to use MySql Workbench to work with a database.
3. How to use MySql Workbench to run SQL Statements.
4. Introduction to Data Types in MySQL.
(Character,Integer,Fixed,Floating,Date,Time,ENUM,SET,Large Objects).

Exp-2

5. Examples on DDL Commands.
CREATE , ALTER, DROP, TRUNCATE a Table.

Exp-3

6. Examples on Implementation of Constraints
PRIMARY KEY,FOREIGN KEY,CHECK,NOT NULL,UNIQUE.

Exp-4

7. Examples on DML & DCL Commands.
INSERT, UPDATE,DELETE, Commands.
COMMIT WORK, ROLLBACK,SAVEPOINT Commands.

EXP-5

8. How to retrieve data from a single table
Select Statement,Select statement with

where clause(Comparison Operators, AND,OR,NOT,IN,BETWEEN,LIKE) ORDER BY clause(sort by column name)

LIMIT Clause

Exp-6

9. Examples on Functions in MySQL.
   String,Numeric,Date,Time and Other Functions.

Exp-7

10. How to retrieve data from two or more tables

    Inner join, self-join, join a table in another db, join more than two tables, outer join, How to use USING, NATURAL Keywords, joins using UNION & SET OPERATORS.

Exp-8

11. Summary Queries:

    Queries using Aggregate functions,Group By Clause,Having Clause

    ROLLUP Operator.

Exp – 9

12. Examples on SUB/SUMMARY Queries Using

    IN, ANY, SOME, ALL , EXISTS Operators

Exp-10

13. Examples on How to
    Create INDEXES,

    Create VIEWS,VIEWS WITH CHECK OPTION
    INSERT,DELETE,DROP on VIEWS

Exp-11

14. Examples on How to

    Create and Call STORED PROCEDURE,
    INPUT,OUTPUT Parameters,
    Validate and Raise errors
    DROP a STORED PROCEDURE.

Exp-12

15.Examples on How to Create ,call and Drop a FUNCTION.

Exp-13

16.Examples on How to Create , View , Alter and Drop a TRIGGER

Learning Resource

Text Books

1. Murach's MySQL by JOEL MURACH, Shroff Publishers & Distributors Pvt.Ltd, June 2012.

III/IV B. TECH. FIRST SEMESTER
MICROPROCESSORS LAB(Required)

Course Code : CS 5L2 **Credits: 2**
Lab Hours: 3 periods/ week **Internal assessment: 25 Marks**
Tutorial:- **Semester end examination: 50 Marks**

Prerequisites: Microprocessor Programming and Interfacing, Computer Networks

Course Objectives:

1. To introduce to students the basics of microprocessor and microcontroller Programming and their applications.
2. The students will be equipped with the basic knowledge of microprocessor and microcontroller interfacing and their applications.

Course Outcomes:

At the end of this course student will:

CO1) Use tools like MSAM/TSAM/Debugger

CO2) Implement assembly code to perform various arithmetic, logical and string operations

CO3) Implement assembly language programs for microprocessor/microcontroller interfaces

Syllabus

a. Microprocessor 8086:

1. Introduction to MASM/TASM/Debugger

2. Arithmetic operation – Multi byte Addition and Subtraction, Multiplication and Division –Signed and unsigned Arithmetic operation, ASCII – arithmetic operation.

3. Logic operations – Shift and rotate – Converting packed BCD to unpacked BCD, BCD to ASCII conversion.
   1. String operation and Instruction prefix: Move Block, Reverse string, Inserting, Deleting, Length of the string, String comparison.

   4. DOS/BIOS programming: Reading keyboard (Buffered with and without echo) – Display characters, Strings.

II. Interfacing:

1.8255-PPI:Write ALP to generate Square wave using PPI.

2. Stepper motor interface with 8086

3. 8279 – Keyboard Display: Write a small program to display a string of characters.

4. 8251 – USART: Write a program in ALP to establish Communication between two processors.

Equipment required for Laboratories:

1. 8086 μP Kits
2. Interfaces/peripheral subsystems I. 8279-KB/Display
II. 8255 PPI
III. 8251 USART IV. Stepper Motor

III/IV B. TECH. FIRST SEMESTER
COMPUTER NETWORKS AND OPERATING SYSTEMS LAB(Required)

Course Code:CS5L3 **Credits: 2**
Lab Hours : 3 periods/ week **Internal assessment: 25 Marks**
Semester end examination: 50 Marks

Prerequisites: Computer Networks, Operating Systems

Computer Networks

Course Objectives:

1. To provide students with a theoretical and practical base in computer networks.
2. To Understand the functionalities of various layers of OSI model

Course Outcomes:

At the end of this course student will:

CO1) Implement DLL functionalities like framing, error detection mechanisms

CO2) Implement basic routing Algorithms

Operating Systems

Course Objectives: to provide students a theoretical and practical base in Operating Systems.

Course Outcomes: At the end of the course students will

CO1) Implement CPU Scheduling algorithms.
CO2) Implement memory management schemes.
CO3)Implement Banker's Algorithm for deadlock Avoidance.

Syllabus:

Part - A

1. Implement the data link layer framing methods such as character, character stuffing and bit stuffing.

2. Implement on a data set of characters the three CRC polynomials – CRC 12, CRC 16 and CRC CCIP.

3. Implement Dijkstra's algorithm to compute the Shortest path thru a graph.

4. Take an example subnet graph with weights indicating delay between nodes. Now obtain Routing table art each node using distance vector routing algorithm

5. Take an example subnet of hosts. Obtain broadcast tree for it.

PART-B

6. Implement CPU Scheduling Algorithms: First Come First Serve & Shortest Job First.
7. Implement CPU Scheduling Algorithms : Priority & Round Robin
8. Memory Management Scheme- I : First Fit & Best Fit
9. Memory Management Scheme-II : FIFO & LRU
10. Implement Banker's Algorithm for deadlock Avoidance.

Learning Resource

Text Books

1. Computer Networks Top Down Approach by Behrouz A Forouzan, Fourth Edition.TMH.
2. Operating Systems Concepts – Abraham Silberchatz, Peter B. Galvin, Greg Gagne , 8th Edition

III/IV B. TECH. FIRST SEMESTER
FREE OPEN SOURCE SOFTWARE TOOLS(Required)

Course Code : CS 5L4 **Credits: 2**
Lab Hours: 3 periods/ week **Internal assessment: 25 Marks**
Tutorial:- **Semester end examination: 50 Marks**

Course Objectives:

To expose students to free open source software environment and introduce them to use open source packages.

Course Outcomes:

At the end of this course student will:

CO1) Implement various applications using build systems

CO2) Understand the installation of various packages in open source operating systems

CO3) Create simple GUI applications using Gambas 3

CO4) Understand various version control sytems

CO5) Understand the kernel configuration and virtual environment

Syllabus:

1. Compiling from source : Learn about the various build systems used like the cmake / make / ant etc. instead of just running the commands.
2. Introduction to package management system : Given set of RPM or DEB, how to build and maintain , serve packages over http or ftp. And also how do you configure client systems to access the package repository.
3. Install various software packages
   b. Install Samba and share files to windows
   c. Install Common Unix Printing System (CUPS)
4. GUI Programming : A Sample programme – Using Gambas 3 since the students have VB knowledge. However, one should try using GTK or QT
5. Version Control System setup and usage using RCS/ CVS/SVN
6. Text Processing with Perl : Simple programs , connecting with database e.g., MYSQL
7. Running PHP : Simple applications like login forms after setting up a LAMP stack
8. Running Python : Some simple exercises

Set up the complete network interface using ifconfig command like setting gateway, DNS, IP tables, etc.

7. Virtualization environment (e.g. Xen / kquemu / lguest / Oracle virtual box) to test an applications, new kernels and isolate applications.

8. Kernel Configuration, Compilation and installation : Download / Access the latest kernel source code from kernel.org, compile the kernel and install it in the local system. Try to view the source code of the kernel.

Learning Resource

Text Books

1. Linux Labs and Open Source Technologies Paperback by Dayanand Ambawade (Author), Deven Shah (Author), Kogent Learning Solutions Inc. (Author)
2. Linux Apache Web Server Administration (Craig Hunt Linux library)Paperback – Import, 23 Nov 2000

III/IV B. TECH. SECOND SEMESTER
ADVANCED JAVA & WEB TECHNOLOGIES(Required)

Course Code: CS 6T1 **Credits: 3**
Lecture: 3 periods/ week **Internal assessment: 30 Marks**
Tutorial: 1period/week **Semester end examination: 70 Marks**

Prerequisites: Java, Database Management Systems,

Course Objectives:

1. To understand the concepts of HyperText Markup Language and Cascading Style Sheets.
2. To learn JavaScript for creating dynamic websites.
3. To learn the operations perform on data among web applications using XML
4. To acquire knowledge on creation of software components using JAVA Beans.
5. To learn Server-Side Programming using Servlets and Java Server Pages.
6. To learn the creation of pure Dynamic Web Application using JDBC.

Course Outcomes:

After completion of this course, the student shall be able to

At the end of this course student will:

CO1) Implement web based applications using features of HTML and XML

CO2) Develop reusable component for Graphical User Interface applications

CO3) Apply the concepts of server side technologies for dynamic web applications

CO4) Implement the web based applications using effective data base access with rich client interaction

Syllabus:

UNIT 1

INTRODUCTION TO WEB TECHNOLOGIES: History of the web, Understanding Web System Architecture, Understanding 3-tier Web Architecture, Overview of HTTP, Introducing HTML document structure, Creating Headings on a web page, Working with links, Creating a Paragraph, Working with images (Hot Spots), Working with tables, working with frames, Introduction to Forms and HTML controls. Inline, External, Internal, Style class, Multiple styles.

UNIT 2

Introducing DHTML, Introducing JavaScript, Client Side benefits of using JavaScript, Embedding JavaScript in an HTML page, Using Variables, Using Operators, Working with Control Flow statements, Working with functions, Handling Events, Using Arrays, Creating objects in JavaScript.

UNIT 3

Introduction to XML, XML Basics, Advanced XML, XML Technologies(DTD),Extensible Style Sheet Transformation (XSLT).

Introducing Java Beans, Introspection Design Patterns for properties, methods, events.

UNIT 4

JDBC Architecture, JDBC Drivers, Communicating with Database using JDBC APIs, Creating a Simple Application , Describing Basic JDBC Statement, Creating tables by using JDBC, Working with Prepared Statement.

Introducing the MVC architecture, Describing Servlets, Understanding Servlets, Introducing the Servlet API, Servlet Life Cycle, Developing First Servlet Application , GenericServlet Class.

UNIT 5

Understanding Request Processing and HTTP Describing the ServletRequest Interface, Working with Initialization Parameters.

Introduction to JSP, Understanding JSP, Describing the JSP Life Cycle, Creating a Simple JSP pages, Working with JSP basic tags and Implicit objects, Working with Java Beans and Action tags in JSP.

Learning Resource

Text Books

1. Web Technologies, Black Book, Kogent Learning Solutions Inc, Dreamtech Press.(UNIT – I , II)
2. JDBC, Servlets, and JSP, New Edition, Santhosh Kumar K , Kogent Learning Solutions Inc, Dreamtech Press (UNIT -III, IV, V)

References

1. Web Technologies , Uttam K. Roy, Volume 2 , Oxford University

2. Core Servlets and Java Server Pages Volume 1 CORE TECHNOLOGIES , Marty Hall and Larry Brown Pearson

3.Java Server Pages, Pekowsky, Pearson.

4.Java Script,D.Flanagan,O'Reilly,SPD.

III/IV B. TECH. SECOND SEMESTER
DESIGN PATTERNS
(Required)

Course Code: CS 6T2 **Credits: 3**
Lecture: 3 periods/ week **Internal assessment: 30 Marks**
Tutorial: 1period/week **Semester end examination: 70 Marks**

Prerequisites: Database Management Systems

Course Objectives:

1. Understand the concept of Design patterns and its importance .
2. Understand the behavioral knowledge of the problem and solutions.
3. Relate the Creational, Structural , behavioral Design patterns.
4. Apply the suitable design patterns to refine the basic design for given context.

Course Outcomes:

CO1) Identify the appropriate design patterns to solve object oriented design problems..

CO2) Develop design solutions using creational patterns.

CO3) Apply structural patterns to solve design problems.

CO4) Construct design solutions by using behavioral patterns.

Syllabus:

UNIT 1

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

UNIT 2

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

UNIT 3

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

UNIT 4

Structural Patterns: Adapter, Bridge, Composite, Decorator, Façade, Flyweight, Proxy.

UNIT 5

Behavioral Patterns: Chain of Responsibility, Command, Interpreter, Iterator, Mediator, Memento, Observer, Strategy, Template Method, Visitor.
Conclusion: What to Expect from Design Patterns, The Pattern Community.

Learning Resource

Text Books

1. Design Patterns By Erich Gamma, Pearson Education
2. Design Patterns Explained By Alan Shalloway,Pearson Education..
3. Meta Patterns designed by Wolf gang , Pearson.

References

1. Head First Design Patterns By Eric Freeman-Oreilly-spd
2. JAVA Enterprise Design Patterns Vol-III By Mark Grand ,Wiley DreamTech.
3. Pattern's in JAVA Vol-I By Mark Grand ,Wiley DreamTech.
4. Pattern's in JAVA Vol-II By Mark Grand ,Wiley DreamTech.

III/IV B. TECH. SECOND SEMESTER
COMPUTER GRAPHICS
(Required)

Course Code: CS6T3 **Credits: 3**
Lecture: 3 periods/ week **Internal assessment: 30 Marks**
Tutorial: 1period/week **Semester end examination: 70 Marks**

Prerequisites: Program Design

Course Objectives:

1. Provide foundation in graphics applications programming
2. Introduce fundamental concepts and theory of computer graphics
3. Give basics of application programming interface (API) implementation based on graphics pipeline approach

Course Outcomes:
At the end of this course student will:

CO1) Understand graphics applications, architectures and openGL program structure.

CO2) Apply basic transformations on objects

CO3) Apply line and polygon clipping algorithms

CO4) Illustrate different projections

CO5) Design interactive programs using openGL

Syllabus:

UNIT 1

Introduction: Applications of computer graphics; A graphics system; Images: Physical and synthetic; Imaging systems; the synthetic camera model; the programmer's interface; Graphics architectures. Graphics Programming: The

Sierpinski gasket; Programming two- dimensional applications. The OpenGL API; Primitives and attributes; Color; Viewing; Control functions; The Gasket program.

UNIT 2

Input and Interaction: Interaction; Input devices; Clients and servers; Display lists; Display lists and modeling; Programming event-driven input; Menus; Picking; Animating interactive programs; Logic operations.

UNIT 3

Geometric Objects and Transformations: Scalars, points, and vectors; Three-dimensional primitives; Coordinate systems and frames; Modeling a colored cube; Affine transformations; Rotation, translation and scaling. Transformations in homogeneous coordinates; Concatenation of transformations; OpenGL transformation matrices.

UNIT 4

Viewing: Classical and computer viewing; Viewing with a computer; Positioning of the camera; Simple projections; Projections in OpenGL; Parallel-projection matrices; Perspective-projection matrices.

UNIT 5

Implementation: Basic implementation strategies; Clipping; Cohen-Sutherland Line-segment clipping; Polygon clipping; Clipping of other primitives; Clipping in three dimensions; Rasterization; Bresenham's algorithm; Polygon rasterization;

Learning Resource

Text Books

1. Interactive Computer Graphics A Top-Down Approach with OpenGL, Edward Angel, 5th Edition, Pearson, 2009.
2. Computer Graphics through OpenGL: From Theory to Experiments, Sumantha Guha, Chapman and Hall/CRC, 2011 (For OpenGL and related examples).

References

1. Computer Graphics with OpenGL, Hearn & Baker, 3rd Edition, Pearson 2004.
2. Computer Graphics Using OpenGL, F.S. Hill, Jr, and M. Kelley, Jr., 3rd Edition, Pearson/PHI, 2009.

III/IV B. TECH. SECOND SEMESTER
DATA WAREHOUSING AND DATA MINING(Required)

Course Code:CS 6T4 **Credits: 3**
Lecture: 3 periods/ week **Internal assessment: 30 Marks**
Tutorial: 1period/week **Semester end examination: 70 Marks**

Prerequisite: Database Management Systems

Course Objectives:

By this course the student will be able to:

1. Understand data mining as a process of knowledge discovery and also about the preprocessing techniques to improve the quality of mining.
2. Learn about Data warehousing and On Line Analytical Processing (OLAP).
3. Understand the kinds of patterns that can be discovered by association rule mining and different classification techniques that builds classifier model for data analysis.
4. Know about the basic concepts of clustering.
5. Evaluate methodological issues underlying the effective application of data mining.

Course Outcomes:

At the end of this course student will:

CO1) Understand the fundamentals of data mining and data warehousing concepts

CO2) Evaluate raw input data and process it for data mining applications

CO3) Discover the interesting patterns from different kinds of databases

CO4) Demonstrate supervised (classification) and unsupervised (clustering) learning techniques

CO5) Describe the outlier detection methods and various data mining methodologies for complex data types

Syllabus:

UNIT 1
Data Warehousing and Online Analytical Processing: Data Warehouse**:** Basic Concepts, Data Warehouse Modeling: Data Cube and OLAP, Data Warehouse Design and Usage, Data Warehouse Implementation.

UNIT 2

Introduction: Fundamentals of data mining: Kinds of data, Data Mining Functionalities, Classification of Data Mining systems, Major issues in Data Mining.

Data Preprocessing: Need for Preprocessing the Data, Data Cleaning, Data Integration, Data Reduction, Data Transformation and Discretization.

UNIT 3

Mining Frequent Patterns, Associations, and Correlations: Basic Concepts, Frequent Item Set Mining Methods.

Classification: Basic Concepts, Decision Tree Induction, Bayes Classification Methods, Rule Based Classification.

UNIT 4

Cluster Analysis: Basic Concepts and Methods, Cluster Analysis, **Partitioning Methods:** k-means and k-mediods**, Hierarchical Method:** Agglomerative Hierarchical clustering (BIRCH**), Density-Based Methods**: DBSCAN, **Grid-based Methods**.

UNIT 5

Outlier Detection: Outliers and Outlier Analysis, Outlier Detection Methods. **Data Mining Trends:** Mining Complex Data Types, Other Methodologies of Data Mining.

Learning Resource

Text Books

1. Data Mining – Concepts and Techniques – 3/e, Jiawei Han , Micheline Kamber & Jian Pei- Elsevier.

References

1. Introduction to Data Mining with Case Studies – 2nd Edition, G.K.Gupta, PHI
2. Introduction to Data Mining: Pang-Ning Tan, Michael Steinbach, Vipin Kumar,Pearson.
3. Data Mining Techniques – ARUN K PUJARI, University Press.
4. Data Warehousing in the Real World, SAM ANAHORY & DENNIS MURRAY, Pearson Edn. Asia.
5. Data Warehousing Fundamentals, PAULRAJ PONNAIAH WILEY STUDENT EDITION.

III/IV B.TECH. SECOND SEMESTER

DATABASE MANAGEMENT SYSTEMS (FREE ELECTIVE)

Course Code : CS6T5FE1 **Credits: 3**
Lecture: 3 periods/ week **Internal assessment: 30 Marks**
Tutorial: 1period/week **Semester end examination: 70 Marks**

Course Objectives:

1. The main objective of this course is to enable students to the fundamental concepts of database analysis and design.
2. To recognize the importance of database analysis and design in the implementation of any Database application and to understand the process of drawing the ER-Diagrams.
3.It also gives the knowledge of the roles of transaction processing and concurrency control.

Course Outcomes:

CO1) Understand the basic principles of database management systems.
CO2) Draw Entity-Relationship diagrams to represent simple database application scenarios
CO3) write SQL queries for a given context in relational database.
CO4)Discuss normalization techniques with simple examples.
CO5)Describe transaction processing and concurrency control concepts.

Syllabus:

UNIT 1

Introduction to Databases: Characteristics of the Database Approach, Advantages of using the DBMS Approach, A Brief History of Database Applications.
Overview of Database Languages and Architectures: Data Models, Schemas and Instances, Three-Schema Architecture and Data Independence, Database Languages and Interfaces, Database System environment, Centralized and Client-Server Architecture for DBMS.

UNIT 2

Relational Model: The Relational Model Concepts , Relational Model Constraints and Relational Database Schemas.
SQL: Data Definition, Constraints, and Basic Queries and Updates,Views(Virtual Tables) in SQL.

UNIT 3

Conceptual Data Modeling : High-Level Conceptual Data Models for Database Design, A Sample Database Application, Entity Types, Entity Sets, Attributes and Keys, Relationship Types, Relationship Sets, Roles, and Structural Constraints, Weak Entity Types.
ER-Diagrams: Refining the ER Design, ER Diagrams, Naming Conventions and Design Issues, Relationship Types of Degree Higher Than Two.

UNIT 4

Database Design Theory: Functional Dependencies, Normal forms based on Primary Keys, Second and Third Normal Forms, Boyce-Codd Normal Form.

UNIT 5

Transaction Processing: Introduction, Transaction and System Concepts, Desirable Properties of Transactions.
Introduction to Protocols for Concurrency Control in Databases : Two-Phase Locking Techniques for Concurrency Control-Types of Locks and System Lock Tables.

Learning Resource

Text Books

1. DATABASE SYSTEMS Models, Languages, Design and Application Programming, 6th Edition, Ramez Elmasri ,Shamkant B.Navathe , Pearson.

III/IV B.TECH. SECOND SEMESTER

WEB TECHNOLOGIES (FREE ELECTIVE)

Course Code : CS6T5FE2 **Credits: 3**
Lecture: 3 periods/ week **Internal assessment: 30 Marks**
Tutorial: 1period/week **Semester end examination: 70 Marks**

Course Objectives:

This course is intended to develop web application using various technologies like HTML,.CSS,JavaScript and JSP.

Course Outcomes:

The students will be able to

CO1) Understand web page creation.
CO2)Develop advanced HTML pages with the help of tags and scripting language.
CO3)Develop user defined tags to exchange the data.
CO4)Understand the object to object communication using JAVA Beans
CO5)Get acquaintance on capabilities of servlet architecture, cookies and session management.
CO6)Understand dynamic content by using JSP architecture and application model.
CO7)Build robust web applications using JSP with JDBC.

Syllabus:

UNIT 1

Introduction to Web Technologies: History of the web, Understanding Web System Architecture, Understanding 3-tier Web Architecture, Overview of HTTP, Web Browsers.
HTML: Introducing HTML document structure, Creating Headings on a web page, Working with links, Creating a Paragraph.

UNIT 2

HTML: Working with images, Working with tables, Working with frames, Introduction to Forms and HTML controls, Introducing Cascading Style sheets..

UNIT 3

Introducing JavaScript: Client Side benefits of using JavaScript, Embedding JavaScript in an HTML page, Using Variables, Using Operators, Working with Control Flow statements, Working with functions, Using Arrays, Creating objects in JavaScript.

UNIT 4

Getting started with Web applications in Java: Introduction to web applications, Exploring Java Based Web Technologies, Introducing Web Architecture models, Introducing MVC architecture. Servlet Life cycle, Creating sample servlet program.

UNIT 5

Working with JSP: Understanding the JSP, Describing the JSP Life Cycle, Creating simple JSP pages, Working with JSP basic tags and Implicit objects.

Working with JDBC: Introduction to JDBC, Exploring JDBC drivers, Describing JDBC APIs, Creating s Simple JDBC application.

Learning Resource

Text Books

1. Web Technologies, Black Book, Kogent Learning Solutions Inc, Dreamtech Press.
2. JDBC, Servlets, and JSP, New Edition, Santhosh Kumar K , Kogent Learning Solutions Inc, Dreamtech Press

References

1. Web Technologies, Uttam K. Roy, Volume 2 , Oxford University
2. Core SERVLETS ANDJAVASERVER PAGES VOLUME 1: CORE TECHNOLOGIES, Marty Hall and Larry Brown Pearson
3. Internet and World Wide Web - How to program, Dietel and Nieto.
4. An Introduction to web Design and Programming -Wang-Thomson.

III/IV B.TECH. SECOND SEMESTER

OBJECT ORIENTED PROGRAMMING THROUGH JAVA (FREE ELECTIVE)

Course Code : CS6T5FE3 **Credits: 3**
Lecture: 3 periods/ week **Internal assessment: 30 Marks**
Tutorial: 1period/week **Semester end examination: 70 Marks**

Course Objectives:

- The main objective of this course is to understand the Object Oriented programming issues in developing more complex software designs. Students will also learn the advantages of Object Oriented programming over the normal and old paradigm structured programming languages. Examples which are demonstrated using java helps the students to understand the concepts and apply the features of Object Oriented programming. The enhancements that are made in the latest certification exams for java are also kept in view. This helps students to keep their skills up to date.

Course Outcomes:

CO1) Understand the key features of the Java programming language
CO2)Apply essential object-oriented programming concepts like dynamic polymorphism, abstract (virtual) methods using Java
CO3)Students will apply the principles behind good object-oriented design.
CO4)Should get exposure to the latest trends in java language and its compatibility in handling numerous complex domains.

Syllabus:

UNIT 1

Java Basics and Anatomy:

Java Basics: OOP's principles, Java History, advantages, Data types, operators, expressions, control statements, methods and recursion, sample programs.

Java Anatomy: Java Objects and References, Constructors, this keyword, Arrays (single and multi-dimensional), String and its immutability, Buffer &Builder Classes, String Tokenizer

UNIT 2

Inheritance (Extending and Implementing): Introduction, Derived Classes, Advantages and Types of Inheritance, Member Accessibility. Overriding, Super, Abstract classes and Methods, Final Classes and Final Methods, Polymorphism, Dynamic Binding.

Interfaces: Differences between classes and interfaces, defining an interface, implementing interface, variables in interface, extending interfaces

UNIT 3

Packaging and Java API

Packages: Defining, Creating and Accessing a Package, importing packages, access controls (public, protected, default, and private). Wrapper Classes and Auto Boxing, I/O classes

UNIT 4

Exception handling and Multithreading: Concepts of exception handling, benefits of exception handling, usage of try, catch, throw, throws and finally, built in exceptions, creating own exception.

Threads: Thread life cycle, creating threads, synchronizing threads

UNIT 5

Graphical User Interaction

Graphical User Interaction: Swings- Introduction, limitations of AWT, components, containers, exploring swing-Frame and JComponent, Icons and Labels, text fields.

Layout managers– border, grid

Event Handling: Events, Event Delegation Model, Event classes, Listeners, handling mouse and keyboard events.

Learning Resource

Text Books

Java Fundamentals, A Comprehensive Introduction, Herbert Schildt, 2014, McGraw-Hill

References

1) Introduction to Java Programming 7/e, Brief version, Y.Daniel Liang, Pearson
2) Java: The complete reference, 7/e, Herbert Scheldt, TMH.
3) Java How to Program, 7/E: Paul Deitel, Deitel & Associates, Inc.

III/IV B.TECH. SECOND SEMESTER

OPERATING SYSTEMS(FREE ELECTIVE)

Course Code : CS6T5FE4 **Credits: 3**
Lecture: 3 periods/ week **Internal assessment: 30 Marks**
Tutorial: 1period/week **Semester end examination: 70 Marks**

Course Objectives:

Course Outcomes:

At the end of this course student will:

CO1) Understand the structure and functionalities of Operating System

CO2) Apply CPU scheduling algorithms, deadlock prevention and detection algorithms and different page replacement algorithms

CO3) Illustrate different problems and solutions related to process synchronization

CO4) Describe the concepts of paging and segmentation for memory management

CO5) Analyze the operating system support for virtual memory, disk management

Syllabus:

UNIT 1

Computer System and Operating System Overview: Operating System Objectives & Functions, Computer System Organization & Architecture, Operating System Structure & Operations. **System Structure:** OS Services, System Calls, Types of system calls.

UNIT 2

Process Management: Process Concept, Process scheduling, Operations on processes, Co-Operating Processes, Interprocess Communication.

Process Scheduling: Basic Concepts, Scheduling Criteria, Scheduling Algorithms (FCFS, SJF, Priority, RR).

UNIT 3

Process Synchronization: Critical Section Problem, Peterson's Solution Synchronization Hardware, Semaphores, Classical problems of synchronization, Monitors.

Deadlocks: System Model, Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance (including Banker's Algorithm), Deadlock Detection & Recovery

UNIT 4

Memory Management: Logical vs. physical address space, Swapping, Contiguous Memory Allocation, Paging, Structure of the Page Table, Segmentation.

Virtual Memory Management: Page fault, Demand Paging, Performance, Page Replacement & its Algorithms (FIFO, LRU Optimal, Clock), Allocation of frames, Thrashing

UNIT 5

Disk Management: Overview of Mass Storage Structure, Disk Scheduling (FCFS, SSTF, SCAN, C-SCAN).

System Calls & IPC: File system related (open, write, read, link, stat, dup, dup2, close), Process related (fork, execve, exit, getpid, getppid, setuid), Directory related (mkdir, chdir, opendir, readdir, closedir), Pipes & FIFOs.

Learning Resource

Text Books

1. Operating System Concepts- Abraham Silberchatz, Peter B. Galvin, Greg Gagne 8th Edition, John Wiley.
2. Advanced UNIX Programming N. B. Venkateswarlu, BS Publications

References

1. Operating Systems' – Internal and Design Principles -- Stallings, Sixth Edition, Pearson education.
2. Operating System Design & Implementation -- Tanenbaum A.S. -- PHI
3. Operating Systems -- Stalling, William -- Maxwell McMillan International Editions.
4. An Introduction to Operating Systems -- Dietel H. N., -- Addison Wesley.
5. Advanced programming in the UNIX environment -- W.Richard Stevens -- pearson Education
6. UNIX and Shell Programming -- Behrouz A. Forouzan, Richard F. Gilberg

III/IV B. TECH. SECOND SEMESTER

ADVANCED JAVA & WEB TECHNOLOGIES LAB(Required)

Course Code: CS 6L1 **Credits: 2**

Lab Hours: 3 periods/ week **Internal assessment: 25 Marks**

Semester end examination: 50 Marks

Prerequisites: Advanced Java & Web Technologies

Course Objectives:

Create a fully functional website to develop an online book store

1. HyperText Markup Language (HTML) and Cascading Style Sheets (CSS) for laying out (formatting) pages that contain text, images and graphics.
2. Extensible Markup Languages (XML is used to store and transport data among webpages), a mechanism for defining new tag sets and interchanging data among web applications.
3. Client-side Programming using JavaScript for validating the data.
4. Creation of software components (objects used for client and server communication) using Beans.
5. Server-Side Programming using servlets are to generate static content and Java Server Pages are used to generate dynamic content.
6. Creating a pure Dynamic Web Application which retrieves the data from Database according to the client request using JDBC.

Course Outcomes:

At the end of this course student will:

CO1) Create and Mange static web pages for given scenario

CO2) Apply server side technologies to establish dynamic applications

CO3) Implement web applications with effective data management

CO4) Develop secure web applications with session management API's

Syllabus:

Exercise 1: To practice HTML fundamental constructs.

1. Headings
2. Links
3. Paragraph
4. Images
5. Tables

Exercise 2: To practice HTML fundamental constructs.

6. Frames
7. Forms and HTML controls

Exercise 3: Project on HTML fundamentals Exercise

4: To practice Cascading style sheets

8. Internal
9. External
10. Inline

Exercise 5: To practice JavaScript and DHTML

11. Simple Programs
12. Handling Events
13. Objects in JavaScript

Exercise 6: Project on DHTML

Exercise 7: To practice XML

- Presenting XML using XSLT

Exercise 8: To practice JDBC connectivity

14. Statement
15. PreparedStatement

Exercise 9: To practice Servlet programming

16.Simple servlet Programming
17.Understanding Life cycle Methods

Exercise 10: To practice Servlet programming

18 Working with initialization parameters
19. RequestDispatcher interface

Exercise 11: To practice JSP

20. JSP basic tags
21. Implicit objects

Exercise 12: To practice JSP

22. Working with Action tags

Learning Resource

Text Books

1.Web Technologies, Black Book, Kogent Learning Solutions Inc, Dreamtech Press.

2. JDBC, Servlets, and JSP, New Edition, Santhosh Kumar K, Kogent Learning Solutions Inc, Dreamtech Press.

III/IV B. TECH. SECOND SEMESTER
UML AND DESIGN PATTERNS LAB(Required)

Course Code: CS 6L2 **Credits: 2**
Lab Hours: 3 periods/ week **Internal assessment: 25 Marks**
Semester end examination: 50 Marks

Prerequisite: UML And Design Patterns

Course Objectives:

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

Course Outcomes:

CO1) Understand the Case studies and design the Model..

CO2) Understand how design patterns solve design problems.

CO3) Develop design solutions using creational patterns.

CO4)Construct design solutions by using structural and behavioural patterns

Exercises:

1. To create a UML diagram of ATM APPLICATION
2. To create a UML diagram of LIBRARY MANAGEMENT SYSTEM
3. Design Abstract factory Design pattern.
4. Design Builder Design pattern.
5. Design Facade Design pattern.
6. Design Bridge Design pattern.
7. Design Decorator Design pattern.

8. User gives a print command from a word document. Design to represent this chain of responsibility design pattern.
9. Design Visitor Design pattern.
10. Design Iterator Design pattern.
11. To design a Document Editor.

Learning Resources

Text Books

1. Grady Booch , James Rumbaugh , Ivar Jacobson : The Unified Modeling Language User Guide, Pearson Education.
2. Design Patterns By Erich Gamma, Pearson Education
3. Meta Patterns designed by Wolf gang , Pearson.

References

1. Design Patterns Explained By Alan Shalloway, Pearson Education..
2. Head First Design Patterns By Eric Freeman - Oreilly-spd.
3. JAVA Enterprise Design Patterns Vol-III By Mark Grand , Wiley Dreamtech.
4. Pattern's in JAVA Vol-I By Mark Grand ,Wiley Dreamtech.
5. Pattern's in JAVA Vol-II By Mark Grand ,Wiley Dreamtech.

III/IV B. TECH. SECOND SEMESTER
COMPUTER GRAPHICS LAB (Required)

Course Code:CS 6L3 **Credits: 2**
Lab Hours: 3 periods/ week **Internal assessment: 25 Marks**
Semester end examination: 50 Marks

Prerequisites: Computer Graphics

Course Objectives:

1. Understand the need of developing graphics application
2. Learn algorithmic development of graphics primitives like: line, circle, polygon etc.
3. Learn the representation and transformation of graphical images and pictures.

Course Outcomes:

At the end of this course student will:

CO1) Draw Geometric primitives using OpenGL

CO2) Execute scan line polygon filling using OpenGL

CO3) Implement basic transformations on objects using OpenGL

CO4) Implement clipping algorithm on lines using OpenGL

Syllabus:

1. Write a program to draw points on a plane in OpenGL
2. Write a program to draw a line on plane in OpenGL.
3. Write a program to draw circle on plane in OpenGL.
4. Write a program draw a white rectangle on a black background in OpenGL.
5. Write a program to draw a square when we click on the mouse button in openGL
6. Write a program to draw a color cube and spin it using open GL transformation matrices in OpenGL.
7. Write a program to create a house like figure and rotate it about a given fixed point using OpenGL functions in OpenGL.
8. Write a program to implement the Cohen-Sutherland line clipping algorithm. Make provision to specify the input line, window for clipping and viewport for displaying the clipped image in OpenGL
9. Write a program to fill any given polygon using scan line area filling algorithm in OpenGL.

Learning Resource

Text Books

Interactive Computer Graphics A Top-Down Approach with OpenGL, Edward Angel, Pearson, 5th Edition, 2009.

III/IV B. TECH. SECOND SEMESTER

SOFT SKILLS COURSE (Required)

Course Code:CS 6L4 **Credits: 1**

Lab Hours: 2 periods/ week **Internal assessment: 25 Marks**

Semester end examination: 50 Marks

Prerequisites : Personality Development Course, Technical English, English Language Communication Skills Lab

Objectives:

1. To introduce them to various aspects of soft skills and personality development.
2. To provide adequate exposure to the process of recruitment.
3. To enhance their communicative competence.
4. To make them ready to face the campus recruitment drives.

Course Outcomes:

Students will be able to

CO1) Improve communication skills.

CO2)Build personality.

CO3)Acquire leadership qualities.

CO4)Improve self confidence.

Syllabus:

1. **Soft skills for career.**
   - Conflict resolution
   - Assertive nature
2. **Soft skills demanded by employer.**
   - Team work
   - Self confidence
   - Responsibility
   - Time management
   - Attitude
   - Empathy
3. **Communication and Networking.**
   - Verbal communication
   - Non verbal communication
4. **Resume preparation.**
5. **Group discussion.**
6. **Interview skills.**

Reference Book:

1. Personality Development and Soft Skills-Baren.K.Mithra Oxford

III/IV B. TECH. SECOND SEMESTER

SEMINAR (Required)

Course Code:CS 6L5 **Credits: 1**

Lab Hours: 2 periods/ week **Internal assessment: ------**

Semester end examination: 50 Marks

Objectives: To get acquainted with the recent technologies and To learn about presentation and communication skills.

Course Outcomes:

CO1) Students will be able to adopt good oral and presentation skills.
CO2) Compile effective power point presentations and report writing.
CO3) Develop better body language.
CO4) Improve knowledge on recent advancements and technologies in IT and ITES

IV/IV B. TECH. FIRST SEMESTER
BIG DATA CONCEPTS
(Required)

Course Code: CS 7T1 **Credits: 3**
Lecture:3 period/week **Internal assessment: 30 Marks**
Tutorial: 1period/week **Semester end examination: 70 Marks**

Prerequisites: Data Structures, File Structures, DBMS, DMDW

Course Objectives:

1. Understand the history of Hadoop and the associated computing techniques.
2. Analyze the Weather Dataset with Unix Tools and Hadoop Tools.
3. Analyze the Hadoop Distributed File system.
4. Evaluate Map Reduce Application development and working process.
5. Analyze the types and formats of Map Reduce.
6. Analyze the Features of Map Reduce.

Course Outcomes:
At the end of this course student will:

CO1) Analyze the data with Hadoop framework

CO2) Explain HDFS concepts, interfaces, and basic file system operations

CO3) Understand the fundamentals of i/o in hadoop

CO4) Develop and implement Map reduce applications on hadoop

CO5) Explore Map reduce types and input formats and output formats

Syllabus:

UNIT 1

Introduction to Hadoop: Data, Data types, Storage and Analysis, Relational Database Management System, Grid Computing, Volunteer Computing, A Brief History of Hadoop, Apache Hadoop and the Hadoop Ecosystems.

Map Reduce: A Weather Dataset: Data Format, Analyzing the data with Unix Tools, Analysing the Data with Hadoop: MapReduce, Java MapReduce, Scaling Out: Data Flow, Combiner Function,s Running a Distributed Map Reduce Job,

UNIT 2

The Hadoop Distributed Filesystem: The Design of HDFS, HDFS Concepts, The Command_Line Interface, Hadoop Filesystems, The Java Interface, Data Flow, Data Ingest with Flume and Sqoop, Parallel Copying with distcp and Hadoop Archieves.

UNIT 3

Developing a Map Reduce Application: The Configuration API: Setting up the Development Environment, Writing a Unit Test with MRUnit, Running Locally on Test Data, Running on a cluster, Tuning a Job, Map Reduce Workflows.

UNIT 4

How Map Reduce Works: Anatomy of a Map Reduce Job Run, Failures, Job Scheduling, Shuffle and Sort, Task Execution.

UNIT 5

Map Reduce Types and Formats: Map Reduce Types, Input Format: Input Splits and Records, Text Input, Binary Input, Multiple Inputs, Database Input and Output, Output Formats: Text Output, Binary Output, Multiple Outputs, Lazy Output, Database Output.

Learning Resource

Text Books

Hadoop: The Definitive Guide, Tom White, 3rd Edition (2012), O'Reilly(SPD).

References

Hadoop Essentials: A Quantitative Approach, Henry H. Liu, 1st Edition (2012), PerfMath Publishers.

IV/IV B. TECH. FIRST SEMESTER
MOBILE APPLICATION DEVELOPMENT (Required)

Course Code: CS 7T2 **Credits: 3**
Lecture:3 periods/week **Internal assessment: 30 Marks**
Tutorial: 1 period/week **Semester end examination: 70 Marks**

Prerequisite: Java, Database Management Systems, Advanced Java and Web Technologies

The main objective of this course is to enable the studetns

- About the Mobile Application programming features.
- To learn the internals of the Android OS
- to learn the Mobile application development using the Android SDK.

Course Outcomes:
At the end of this course student will able

CO1) To understand the key features of various Mobile Operating Systems (specially Android)

CO2) To know essential Android programming concepts

CO3) To develop Android Applications using GUI Components.

CO4) To demostrate and implement DataBase Connectivity Applications.

Syllabus:

UNIT 1

Android Introduction and Basics: Introduction to Android Platform, Android vs. other mobile platforms, Android Stack, Android Versions and Installing Android SDK components, updating SDK components,Android emulator, Sample programs on emulator.

Java role and java for Android: Reshaping client side java as Android, java type system, scope and idioms of java programming.

UNIT 2

Android Applications and its Anatomy: Android programming model vs. traditional programming models, Activities, Intents and Tasks, Other Android Components, Component Life Cycles, Static Application Resources and Context Android Application Runtime Environment: Activity life cycle, Manifest File, Layout XML Code, Strings, The R File.

UNIT 3

Android Frame Work and User Interface Design: Android GUI Architecture, Assembling a Graphical Interface, different layouts – Linear Layout and Table Layout etc., Drawable Resources, Drawable Resources, Resolution and density independence Working with common widgets, List View and Adapters, The Menu and the Action Bar, View Debugging and Optimization.

UNIT 4

Fragments : Creating a Fragment, Fragment Life Cycle, Fragment Manager, Fragment Transactions The Support Package, Fragments and Layout.

Content Providers: Understanding Content Providers, Defining a Provider Public API, Writing and Integrating a

Content Provider, File Management and Binary Data, Android MVC & Content observation. Sample Content Provider.

UNIT 5

Handling and Persisting Data: Relational Database Overview, SQLite, SQL and the Database-Centric Data Model for Android Applications, the Android Database Classes, Database Design for Android Applications.

Learning Resource

Text Books

1. Programming Android, 2nd Edition(Oct-2012), by Zigurd Mednieks, Larid Dornin, G.Blake Meike, Masumi Nakamura , O'reilly (SPD) Publications.

References

1. Beginning Android 4 Application Development, by *Wei-Meng Lee* **,** Wiley India

2.Beginning Android 4, (2012) , by *Grant Allen* , Apress publications.
3.Android Application Development (programming with Google SDK), by Rick Rogers, Jhon Lombarado, Zigurd M

IV/IV B. TECH. FIRST SEMESTER
INFORMATION SECURITY(Required)

Course Code: CS 7T3 **Credits: 3**
Lecture:3 periods/week **Internal assessment: 30 Marks**
Tutorial: 1period/week **Semester end examination: 70 Marks**

Prerequsites: Program Design, Computer Graphics

Course Objectives:

The main goal of this course is to provide background, foundation, and insight into the many dimensions of information security. This knowledge will serve as basis for further deeper study into selected areas of the field, or as an important component in further studies and involvement in computing as a whole. The primary objectives of the course are as follows. Understand information security's importance in our increasingly computer-driven world. Master the key concepts of information security and how they "work." The course will be organized around a few broad themes:

1. Foundations: security mindset, essential concepts (policy, CIA, etc.)
2. Software security: vulnerabilities and protections, malware, program analysis
3. Practical cryptography: encryption, authentication, hashing, symmetric and asymmetric crypto Networks: wired and wireless networks, protocols, attacks and countermeasures.

Course Outcomes:

At the end of this course student will:

CO1) Understand the need of security over the network and define the cryptographic mechanism

CO2) Apply appropriate symmetric key algorithms

CO3) Apply appropriate Asymmetric key algorithms

CO4) Attribute the security mechanism in network transmission

CO5) Understand various vulnerabilities on system security

Syllabus:

UNIT 1
Security Attacks (Interruption, Interception, Modification and Fabrication), Security Services (Confidentiality, Authentication, Integrity, Non-repudiation, access Control and Availability) and Mechanisms, A model for Internetwork security, Internet Standards and RFCs.

UNIT 2
Conventional Encryption Principles, Conventional encryption algorithms, cipher block modes of operation, location of encryption devices, key distribution Approaches of Message Authentication, Secure Hash Functions and HMAC.

UNIT 3
Public key cryptography principles, public key cryptography algorithms: RSA,Diffie-Hellmen key exchange algorithms, digital signatures, digital Certificates, Certificate Authority and key management Kerberos, X.509 Directory Authentication Service. Email privacy: Pretty Good Privacy (PGP) and S/MIME.

UNIT 4
IP Security Overview, IP Security Architecture, Authentication Header, Encapsulating Security Payload, Combining Security Associations and Key Management. Web Security Requirements, Secure Socket Layer (SSL) and Transport Layer Security (TLS), Secure Electronic Transaction (SET).

UNIT 5
Basic concepts of SNMP, SNMPv1 Community facility and SNMPv3. Intruders, Viruses and related threats. Firewall Design principles, Trusted Systems. Intrusion Detection Systems.

Learning Resource

Text Books

1. Network Security Essentials (Applications and Standards) by William Stallings Pearson Education.

References

1. Hack Proofing your network by Ryan Russel, Dan Kaminsky, Rain Forest Puppy, Joe Grand, David Ahmad, Hal Flynn Ido Dubrawsky, Steve W. Manzuik and Ryan Permeh, Wiley Dreamtech.
2. Fundamentals of Network Security by Eric Maiwald (Dreamtech press)
3. Cryptography and network Security, Third edition, Stallings, PHI/Pearson
4. Principles of Information Security, Whitman, Thomson.5. Introduction to Cryptography, Buchmann, Springer.

IV/IV B. TECH. FIRST SEMESTER
CLOUD COMPUTING (Elective-I)

Course Code: CS 7T4A **Credits: 3**
Lecture:3 periods/week **Internal assessment: 30 Marks**
Tutorial: 1period/week **Semester end examination: 70 Marks**

Prerequisites: Discrete Mathematics, Computer Networks

Course Objectives:

1. To understand the concepts of Cloud Computing.
2. To learn Taxonomy of Virtualization Techniques.
3. To learn Cloud Computing Architecture.
4. To acquire knowledge on Aneka Cloud Application Platform.
5. To learn Industry Cloud Platforms.

Course Outcomes:
At the end of this course student will:

CO1) Understand the concept of virtualization and how this has enabled the development of Cloud Computing

CO2) Know the fundamentals of cloud, cloud Architectures and types of services in cloud

CO3) Understand scaling, cloud security and disaster management

CO4) Design different Applications in cloud

CO5) Explore some important cloud computing driven commercial systems

Syllabus:

UNIT 1
Introduction to Cloud: Cloud Computing at a Glance, The Vision of Cloud Computing, Defining a Cloud, A Closer Look, Cloud Computing Reference Model. Characteristics and Benefits, Challenges Ahead, Historical Developments.

Virtualization: Introduction, Characteristics of Virtualized Environment, Taxonomy of Virtualization Techniques, Virtualization and Cloud computing, Pros and Cons of Virtualization, Technology Examples- VMware and Microsoft Hyper-V.

Before the Move into the Cloud: Know Your Software Licenses, The Shift to a Cloud Cost Model, Service Levels for Cloud Applications.

UNIT 2

Cloud Computing Architecture : Introduction, Cloud Reference Model, Architecture, Infrastructure / Hardware as a Service, Platform as a Service, Software as a Service, Types of Clouds, Public Clouds, Private Clouds, Hybrid Clouds, Community Clouds, Economics of the Cloud, Open Challenges, Cloud Interoperability and Standards, Scalability and Fault Tolerance.

Ready for the Cloud: Web Application Design, Machine Image Design, Privacy Design, Database Management, Data Security, Network Security, Host Security, Compromise Response.

UNIT 3

Defining the Clouds for Enterprise: Storage as a service, Database as a service, Process as a service, Information as a service, Integration as a service and Testing as a service.

Scaling a cloud infrastructure - Capacity Planning, Cloud Scale.

Disaster Recovery: Disaster Recovery Planning, Disasters in the Cloud, Disaster Management.

UNIT 4

Aneka: Cloud Application Platform Framework Overview, Anatomy of the Aneka Container, From the Ground Up: Platform Abstraction Layer, FabricServices, FoundationServices, ApplicationServices, Building Aneka Clouds, InfrastructureOrganization, LogicalOrganization, Private Cloud Deployment Mode, Public Cloud Deployment Mode,Hybrid Cloud Deployment Mode, Cloud Programming and Management,Aneka SDK, Management Tools.

UNIT 5

Cloud Applications: Scientific Applications – Health care, Geoscience and Biology. Business and Consumer Applications- CRM and ERP, Social Networking, Media Applications and Multiplayer Online Gaming.

Cloud Platforms in Industry: Amazon Web Services- Compute Services, Storage Services, Communication Services and Additional Services. Google AppEngine-Architecture and Core Concepts, Application Life-Cycle, cost model. Microsoft Azure- Azure Core Concepts, SQL Azure.

Learning Resource

Text Books

1. Mastering Cloud Computing by Rajkumar Buyya, Christian Vecchiola, S.Thamarai Selvi from TMH 2013.
2. George Reese Cloud Application Architectures, First Edition, O'Reilly Media 2009.

References

1. Cloud Computing and SOA Convergence in Your Enterprise *A Step-by-Step Guide* by David S. Linthicum from Pearson 2010.
2. Cloud Computing 2nd Edition by Dr. Kumar Saurabh from Wiley India 2012.
3. Cloud Computing – web based Applications that change the way you work and collaborate Online – Micheal Miller.Pearson Education.

IV/IV B. TECH. FIRST SEMESTER
ADVANCED DATABASES (Elective-I)

Course Code: CS 7T4B **Credits: 3**
Lecture:3 periods/week **Internal assessment: 30 Marks**
Tutorial: 1period/week **Semester end examination: 70 Marks**

Prerequisites: Databases _ Algorithms and data structures _ Programing and programming engineering

Course Objectives:

1. The main objective of this course is to familiarize students with the Advanced concepts on data bases.
2. To Comprehend the concepts of Query Processing & Optimization Techniques.
3. It also gives the knowledge of transaction processing and concurrency control.
4. To Introduce the Recovery Techniques.
5. To Introduce the basic concepts of Emerging and Advanced Databases .

Course Outcomes:
At the end of this course student will:

CO1) Understand the Processing and Optimization of Queries

CO2) Interpret the transaction management Techiniques

CO3) Discuss Concurrency control Protocols

CO4) Describe different Database Recovery Techniques

CO5) Summarize the relevant Emerging and Advanced Database concepts

Syllabus:

UNIT 1

Introduction to Query Processing and Query Optimization Techniques:

Translating SQL Queries into Relational Algebra, Algorithms for External Sorting, Algorithms for SELECT and JOIN Operations, Algorithms for PROJECT and Set Operations, Implementing Aggregate Operations and OUTER JOINs, Combining Operations Using Pipelining, Using Heuristics in Query Optimization, Using

Selectivity and Cost Estimates in Query Optimization, Overview of Query Optimization in Oracle, Semantic Query Optimization.

UNIT 2

Foundations of Database Transaction Processing: Introduction to Transaction and System Concepts, Desirable Properties of Transactions, Characterizing Schedules Based on Recoverability, Characterizing Schedules Based on Serializability, Transactions Support in SQL.

UNIT 3

Introduction to Protocols for Concurrency Control in Databases : Two-Phase Locking Techniques for Concurrency Control, Concurrency Control Based on Timestamp Ordering, Multiversion Concurrency Control Techniques, Validation Concurrency Control Techniques, Granularity of Data Items and Multiple Granularity Locking, Using Locks for Concurrency Control in Indexes, Other Concurrency Control Issues.

UNIT 4

Introduction to Database Recovery Protocols: Recovery Concepts, NO_UNDO/REDO Recovery Based on Deferred Update, Recovery Techniques Based on Immediate Update, Shadow Paging, The ARIES Recovery Algorithm, Recovery in multidatabase Systems, Database Backup and Recovery from Catastrophic Failures.

UNIT 5

Emerging Database Technologies and Applications : Mobile Data Management, Multimedia Data Management, Geographic Information Systems(GIS),Biological and Genomic Databases and Emerging Applications.

Advanced Database Models and Applications : Active Database Concepts and Triggers, Temporal Database Concepts, Spatial Database Concepts, Multimedia Database Concepts, Introduction to Deductive Databases.

Learning Resource

Text Books

DATABASE SYSTEMS Models, Languages, Design and Application Programming, 6th Edition, Ramez Elmasri ,Shamkant B.Navathe , Pearson.

References

1.Data base System Concepts, 5th Edition, Abraham Silberschatz, Henry F Korth, S.Sudarshan,Mc Graw Hill.

2.Data base Management Systems, 3rd Edition, Raghurama Krishnan, Johannes Gehrke, TMH

IV/IV B. TECH. FIRST SEMESTER
IMAGE PROCESSING (Elective-I)

Course Code: CS 7T4C **Credits: 3**
Lecture:3 periods/week **Internal assessment: 30 Marks**
Tutorial: 1period/week **Semester end examination: 70 Marks**

Prerequisites: signals and systems

Course Objectives:

1. To introduce students to the Basic concepts and analytical methods of analysis of digital images.
2. To Study fundamental concepts of Digital Image Processing and basic relations among pixels.
3. To Study different Spatial and Frequency domain concepts.
4. To understand Restoration process of degraded image and Multi resolution processing.
5. To understand image compression and Segmentation Techniques.

Course Outcomes:

At the end of this course student will:

CO1) Understand different components of image processing system

CO2) Describe various image transforms, enhancement techniques using various processing methods

CO3) Illustrate the compression and segmentation techniques on a given image

CO4) Demonstrate the filtering and restoration of images(pixels) with examples

CO5) Illustrate the various schemes for image representation and edge detection techniques with examples

Syllabus:

UNIT 1

Introduction: Digital Image Processing, Fundamental Steps in Digital Image Processing, Components of an Image Processing System.

Digital Image Fundamentals: Elements of Visual Perception, Image Sensing and Acquisition, Image Sampling and Quantization, Some basic Relationships between Pixels.

UNIT 2

Image Enhancement in the Spatial Domain: Some Basic Gray Level Transformation, Histogram Processing, Enhancement Using Arithmetic/Logic Operations, Basics of Spatial Filtering, Smoothing spatial Filters, Sharpening spatial Filters.

Image Enhancement in the Frequency Domain: Introduction to the Fourier Transform and the Frequency Domain, Smoothing frequency-domain Filters, Sharpening Frequency-domain Filters, Homomorphic Filtering, Implementation.

UNIT 3

Image Restoration: A Model of the Image Degradation/Restoration Process, Linear, Position-Invariant Degradations, Inverse Filtering, Minimum Mean Square Error (Wiener) Filtering, Constrained Least Squares Filtering. Wavelets and Multi resolution Processing: Multi resolution Expansions, Wavelet Transforms in one Dimension, The Fast Wavelet Transform, Wavelet Transforms in Two Dimensions.

UNIT 4

Image Compression: Image Compression Models, Error-free Compression, Lossy Compression, Image Compression Standards.

Image Segmentation: Detection of Discontinuities, Edge Linking and Boundary Detection, Thresholding, Region-Based Segmentation.

UNIT 5

Representation and Description: Various schemes for representation, boundary descriptors, and regional descriptors.

Learning Resource

Text Books

1. Rafael C. Gonzalez and Richard E. Woods, Digital Image Processing. Prentice Hall India/Pearson Education.

References

1. A.K.Jain, Fundamentals of Digital Image Processing. Prentice Hall India.
2. Madhuri.A.Joshi, Digital Image Processing, PHI.
3. Sonka, Image Processing, Analysis and Machine Vision. Cengage Publications.
4. Fundamentals of Digital Image Processing, Anna durai,Shanmuga lakshmi.

IV/IV B. TECH. FIRST SEMESTER
ADVANCED COMPUTER ARCHITECTURE (Elective-I)

Course Code: CS 7T4D **Credits: 3**
Lecture:3 periods/week **Internal assessment: 30 Marks**
Tutorial: 1period/week **Semester end examination: 70 Marks**

Prerequisites: Programming and Data structures, Discrete Maths, and a basic knowledge of Computer organization.

Course Objectives:

1. Understand the Concept of Parallel Processing and its applications.
2. Implement the Hardware for Arithmetic Operations.
3. Analyze the performance of different scalar Computers.
4. Develop the Pipelining Concept for a given set of Instructions.
5. Distinguish the performance of pipelining and non pipelining environment in a processor.

Course Outcomes:

At the end of this course student will:

CO1) Understand the Concept of Parallel Processing and its applications

CO2) Implement the Hardware for Arithmetic Operations

CO3) Analyze the performance of different scalar Computers

CO4) Develop the Pipelining Concept for a given set of Instructions

CO5) Distinguish the performance of pipelining and non pipelining environment in a processor

Syllabus:

UNIT 1
Pipeline and vector processing : Parallel Processing, Pipelining, Arithmetic Pipeline, Instruction Pipeline, RISC Pipeline, Vector Processing, Array Processors.

UNIT 2
Computer Arithmetic : Addition and Subtraction, Hardware Implementation, Multiplication Algorithms and Hardware Implementation, Division Algorithms and Hardware Implementation, Floating Point Arithmetic Operations.

UNIT 3

Parallel Computer Models : Evolution of Computer Architecture, System Attributes to Performance, Shared Memory Multiprocessors, Distributed Memory Multicomputers, Vector Super Computers, SIMD Super Computers.

UNIT 4

Processors and Memory Hierarchy : Advanced Processor Technology: Design Space of Processors, Instruction-Set Architectures, CISC scalar Processors, RISC scalar Processors, Super Scalar and Vector Processors: Superscalar Processors.

UNIT 5

Pipelining and Superscalar Techniques : Linear Pipeline Processors: Asynchronous and Synchronous models, Clocking and Timing Control, Speedup, Efficiency and Throughput, Pipeline Schedule Optimization, Instruction Pipeline Design: Instruction Execution Phases, Mechanisms for Instruction Pipelining, Dynamic Instruction Scheduling, Branch Handling Techniques.

Learning Resource

Text Books

1. Computer System Architecture, Morris M. Mano, 3rd edition, Pearson/Prentice Hall India.
2. Advanced Computer Architecture, Kai Hwang, McGraw-Hill, India.

References

1.Computer Organization and Achitecture, William Stallings ,8th edition,PHI
2.Computer Organization, Carl Hamachar, Vranesic,Zaky, 5th edition, McGraw Hill.

IV/IV B. TECH. FIRST SEMESTER
SOFTWARE ENGINEERING (Elective-II)

Course Code: CS 7T5A	**Credits: 3**
Lecture:3 periods/week	**Internal assessment: 30 Marks**
Tutorial: 1period/week	**Semester end examination: 70 Marks**

Prerequisites: Data structures, Algorithms

Course Objectives:

1. An understanding of different software processes and how to choose between them
2. How to elicit requirements from a client and specify them
3. Designing the large, including principled choice of software architecture, the use of modules and interfaces to enable separate development, and design patterns.
4. Understanding good coding practices, including documentation, contracts, regression tests and daily builds.

Course Outcomes:

At the end of this course student will:

CO1) Understand the core principles of software engineering

CO2) Apply appropriate software process model for a given scenario

CO3) Analyze the requirements for a given problem

CO4) Apply the design paradigms to design simple software system

CO5) Identify the fundamental principle of test-driven development methods

CO6) Interpret the risk strategies to assure the quality of software

Syllabus:

UNIT 1

Software and Software Engineering: The Nature of Software, the Unique Nature of Webapps, Software Engineering, the Software

Process, Software Engineering Practice, Software Myths

Process Models: Generic Process Model, Prescriptive Process Models, Specialized Process Models, Unified Process

UNIT 2

Understanding Requirements: Eliciting Requirements, Developing Use Cases. **Requirements Modelling:** Scenario Based Modelling, Class Based Modelling

UNIT 3

Design Concepts: Design Process, Design Concepts, And The Design Model. **Architectural Design:** Architectural Styles, Architectural Design, **Component Level Design:** Designing Class Based Components

UNIT 4

Software Testing Strategies: A Strategic Approach to Software Testing, Test Strategies for Conventional Software, Test Strategies for Object Oriented Software, Validation Testing, System Testing, the Art of Debugging.

Testing Conventional Applications: White Box Testing, Black-Box Testing.

UNIT 5

Risk Management: Reactive Vs. Proactive Risk Strategies, Software Risks, Risk Identification, Risk Projection, Risk Refinement, RMMM, RMMM Plan

Quality Management: What Is Quality, Software Quality

Software Quality Assurance: Elements Of Software Quality Assurance, SQA Tasks, Goals And Metrics, The ISO 9000 Quality Standard, SQA Plan.

Learning Resource

Text Books

Software Engineering, 7/E , Roger S. Pressman , TMH

References

1. Software Engineering, A Precise Approach, Pankaj Jalote, Wiley
2. Software Engineering Principles and Practice, W S Jawadekar, TMH
3. Software Engineering Concepts, R Fairley, TMH.

IV/IV B. TECH. FIRST SEMESTER
EMBEDDED SYSTEMS (Elective-II)

Course Code: CS 7T5B **Credits: 3**
Lecture:3 periods/week **Internal assessment: 30 Marks**
Tutorial: 1period/week **Semester end examination: 70 Marks**

Prerequisites: **C Language, I/O, Analog and Digital interfacing, and peripherals.**

Course Objectives:

The objective of this course is to equip the students with the basic concepts of embedded system, applications in which they are used, 8051 microcontroller programming concepts and various aspects of embedded system design from Hardware and Software points of view and it describes tools and methodologies needed for embedded system design. It provides RTOS concepts for coding the embedded system software routines. It tells what makes a system a real-time system and describes the characteristics of latency in real-time systems.

Course Outcomes:

At the end of this course student will:

CO1) Understand the microprocessor architecture and its components used in embedded systems

CO2) Write the 8051 assembly language code for specific purposes

CO3) impliment code for interfacing various devices.

CO4) Develop simple embedded systems for real time operations

CO5) Compose simple embedded system with error free software to obtain target system

Syllabus:

UNIT 1

Embedded Systems Basics: Introduction to Embedded systems, Examples of embedded systems, TypicalHardware,Gates,TimingDiagrams,Memory, Microprocessors, Buses, DirectMemoryAccess, Interrupts, Microprocessor Architecture, and Interrupt Basics.

UNIT 2

The 8051 Architecture : Introduction, 8051 Micro controller Hardware, Input/output Pin Ports and Circuits, External Memory, Serial data Input/output, Interrupts.

UNIT 3

Basic Assembly Language Programming Concepts: The Assembly Language Programming Process, Programming Tools and Techniques, Programming the 8051.

UNIT 4

Moving Data: Introduction, Addressing Modes, External Data Moves, Code Memory Read-Only Data Moves, Push and Pop Opcodes, Data Exchanges.

Basic Design Using a Real-Time Operating System: Message Queues, Mailboxes and Pipes, Timer Functions, Events, Memory Management, Interrupt Routines in an RTOS Environment

UNIT 5

Applications: Introduction, keyboards, Human Factor, Key Switch Factors, Keyboard Configurations, Displays, Seven-Segment Numeric Display, D/A and A/D Conversions.

Embedded Software Development Tools: Host and Target machines, Linker/Locators for Embedded Software, Getting Embedded Software into the Target System; Debugging Techniques: Testing on Host Machine, Using Laboratory Tools, An Example System.

Learning Resource

Text Books

1.An Embedded Software Primer, David E. Simon, Pearson Education.
2.The 8051 Microcontroller, Third Edition, Kenneth J.Ayala, Thomson.

References

1. 8051 Microcontrollers, Satish Shah, Oxford Higher Education.
2. Embedded Microcomputer Systems Real Time Interfacing, Jonathan W.Valvano, Cengage Learning.
3. Micro Controllers, Ajay V Deshmukhi, TMH.
4. Embedded System Design, Frank Vahid, Tony Givargis, John Wiley.
5. Microcontrollers, Raj kamal, Pearson Education.

a. http://nptel.ac.in/courses.php
b. http://jntuk-coeerd.in/

IV/IV B. TECH. FIRST SEMESTER
DISTRIBUTED SYSTEMS (Elective-II)

Course Code: CS 7T5C **Credits: 3**
Lecture:3 periods/week **Internal assessment: 30 Marks**
Tutorial: 1period/week **Semester end examination: 70 Marks**

Prerequisites: Discrete Mathematics, Computer Networks.

Course Objectives:

The main objective of this course is to understand the hardware and software issues in modern distributed systems. Students will also learn distributed architecture, naming, synchronization, consistency and replication, fault tolerance, security, and distributed file systems. Examples from current popular distributed systems such as peer-to-peer (P2P) systems will be analyzed.

Course Outcomes:
At the end of this course student will:

CO1) Understand the conceptual model of a distributed system and its architectural models

CO2) Exemplify Inter process communication mechanisms

CO3) Suggest appropriate algorithm for a given application in Distributed Systems

CO4) Outline various Consistency Models and replication management techniques with examples

CO5) Understand various mechanisms for software fault tolerance & security in Distributed Systems.

Syllabus:

UNIT 1

Introduction Of Distributed System: Goals, Types of Distributed systems**.**

Architectures: Architectural Styles, System architectures, Self management in distributed systems.

UNIT 2

Processes: Threads, Virtualization, Clients, Servers, Code Migration, Software Agents.

Communication: Fundamentals, Remote Procedure Call, Message Oriented Communication,

Stream-Oriented Communication, Multicast Communication**.**

UNIT 3

Naming: Names, Identifiers and Addresses, Flat Naming, Structured Naming, Attribute-Based Naming

Synchronization: Clock Synchronization, Logical Clocks, Mutual Exclusion, Global Positioning of nodes, Election Algorithms.

UNIT 4

Consistency and Replication: Introduction, Data-Centric Consistency Models, Client Centric Consistency Models, Replica Management, Consistency Protocols, Examples.

UNIT 5

Fault Tolerance: Introduction to Fault Tolerance, Process Resilience, Reliable Client-Server Communication, Reliable Group Communication, Distributed Commit, Recovery.

Security: Introduction, Secure channels, Access Control, Security Management

Learning Resource

Text Books

Distributed Systems – Principles and Paradigms, Andrew S. Tanenbaum, Maarten Van Steen, 2/e, PHI.

References

1. Distributed Systems Concepts and Design, George Couloris, Jean Dollimore, Tim Kindberg, Gordan Blair, 4/e, PEARSON.
2. Distributed Operating Systems Concepts and Design, Pradeep K. Sinha, PHI.

IV/IV B. TECH. FIRST SEMESTER
WEB SERVICES (Elective-II)

Course Code: CS 7T5D **Credits: 3**
Lecture:3 periods/week **Internal assessment: 30 Marks**
Tutorial: 1period/week **Semester end examination: 70 Marks**

Prerequisites: XML, HTTP, TCP/IP concepts.

Course Objectives:

1. To Understand Web Services and implementation model for SOA
2. To Understand the SOA, its Principles and Benefits
3. To Understand XML concepts
4. To Understand paradigms needed for testing Web Services
5. To explore different Test Strategies for SOA-based applications

Course Outcomes:

At the end of this course student will:

CO1) Understand the principlesof SOA

CO2) Efficiently use market leading environment tools to create and consume web services

CO3) Indentify and select the appropriate framework components in creation of webservice solution

CO4) Apply OOP principles to creation of webservice solutions.

Syllabus:

UNIT 1

Evolution and Emergence of Web Services – Evolution of distributed computing. Core distributed computing technologies – client/server, CORBA, JAVA RMI, Micro Soft DCOM, MOM, Challenges in Distributed Computing, Introduction to Web Services – The definition of web services, basic operational model of web services, tools and technologies enabling web services, benefits and challenges of using web services.

UNIT 2

Web Service Architecture – Web services Architecture and its characteristics, core building blocks of web services, standards and technologies available for implementing web services, web services communication, basic steps of implementing web services.

UNIT 3

Brief Over View of XML – XML Document structure, XML namespaces, Defining structure in XML documents, Reuse of XML schemes, Document navigation and transformation. SOAP : Simple Object Access Protocol, Inter-application communication and wire protocols, SOAP as a messaging protocol, Structure of a SOAP message, SOAP envelope, Encoding, Service Oriented Architectures, SOA revisited, Service roles in a SOA, Reliable messaging,

UNIT 4

Describing Web Services – WSDL introduction, non functional service description, WSDL1.1 Vs WSDL 2.0, WSDL document, WSDL elements, WSDL binding, WSDL tools, WSDL port type, limitations of WSDL.

UNIT 5

Registering and Discovering Services : The role of service registries, Service discovery, Universal Description, Discovery, and Integration, UDDI Architecture, UDDI Data Model, Interfaces, UDDI Implementation,

Learning Resource

Text Books

1. Web Services & SOA Principles and Technology, Second Edition, Michael P. Papazoglou.
2. Developing Java Web Services, R. Nagappan, R. Skoczylas, R.P. Sriganesh, Wiley India.
3. Developing Enterprise Web Services, S. Chatterjee, J. Webber, Pearson Education.

References

1. XML, Web Services, and the Data Revolution, F.P.Coyle, Pearson Education.
2. Building web Services with Java, 2nd Edition, S. Graham and others, Pearson Education.
3. Java Web Services, D.A. Chappell & T. Jewell, O'Reilly, SPD.
4. McGovern, et al., "Java web Services Architecture", Morgan Kaufmann

Publishers, 2005.

IV/IV B. TECH. FIRST SEMESTER
DATA ANALYTICS LAB(Required)

Course Code: CS7L1 **Credits: 2**
Lecture:-- **Internal assessment: 25 Marks**
Lab: 3period/week **Semester end examination: 50 Marks**

Prerequisites: Big Data Concepts

At the end of this course student will:

CO1) Install and run Hadoop in standalone mode, pseudo mode and fully distributed cluster environment.

CO2) Develop Hadoop Mapreduce algorithms

CO3) Calculate basic analytics using Hadoop and Mapreduce.

Syllabus:

Getting Hadoop Up and Running in a cluster:

1. Setting up Hadoop on standalone machine.
2. Wordcount Map Reduce program using standalone Hadoop.
3. Adding the combiner step to the Wordcount Map Reduce program.
4. Setting up HDFS.
5. Using HDFS monitoring UI
6. HDFS basic command-line file operations.
7. Setting Hadoop in a distributed cluster environment.
8. Running the WordCount program in a distributed cluster environment.
9. Using Map Reduce monitoring UI

Hadoop Map Reduce Applications:

1. Choosing appropriate Hadoop data types.
2. Implementing a custom Hadoop Writable data type.
3. Implementing a custom Hadoop key type.
4. Emitting data of different value types from a mapper.
5. Choosing a suitable Hadoop Input Format for your input data format.
6. Formatting the results of Map Reduce Computation – using Hadoop Output Formats.

Analytics

1. Simple analytics using Map Reduce.

2. Performing Group-By using Map Reduce.
3. Calculating frequency distributions and sorting using Map Reduce.
4. Plotting the Hadoop results using GNU plot.
5. Calculating histograms using Map Reduce.
6. Calculating scatter plots using Map Reduce.
7. Parsing a Complex dataset with Hadoop.
8. Joining two datasets using Map Reduce

Learning Resource

Text Books

Hadoop Map Reduce Cookbook, Srinath Perera & Thilina Gunarathne, 2013, PACKT PUBLISHING.

IV/IV B. TECH. FIRST SEMESTER
MOBILE APPLICATION DEVELOPMENT LAB (Required)

Course Code: CS 7L2 **Credits: 2**
Lecture:-- **Internal assessment: 25 Marks**
Lab: 3 period/week **Semester end examination: 50 Marks**

Prerequisite: Mobile Application Development

Course Outcomes:

At the end of this course student will:

CO1) Apply essential Android Programming concepts.

CO2) Develop various Android applications related to layouts & rich uses interactive interfaces

CO3) Develop Android applications related to mobile related server-less database like SQLITE

Syllabus:

Toast msgs

1. Write an android program to implement activity life cycle using toast messages with proper positioning.

Lay Outs

2. Write an android program to print the set of alphabets/strings in a linear layout and in table layout.
   a. Write an android program to align text boxes labels, buttons in a Emulator using relative and linear layout tags in a layout.xml.

Dialogs and Menu

3. Write an android program to demonstrate DatePickerDialog, TimePickerDialog with current date and current running time.

   a. Write an android program to demonstrate a Menu with name File with New and Open as menu items. Give toast msgs on click of each menu item. (if possible implement the content in 3.a in one tab and other set of items in another tab.)

4. Write an android program to switch from one activity to another using Intent. When the activity is changed disable the use of back button to avoid going to previous activity

Views

5. Write an android program to demonstrate scroll view and list view.

(List view should array adapter. The adapter should use array list of companies.Each item in the list view should have company name, company address and its annual revenue.)

SQLite Database

6. Write an android program to implement the following operations using SQLite Database.

- Create the SQLite Database Object.
- Execute the CRUD Operations required for the application
- Close the database.

Case Study

7. Divide students into batches and suggest them to develop any interested project such as.:

a. Student Mark Entry System

b. Enquiry System

c. Monitoring System

Learning Resource

References

Android Cook Book, by Ian F. Darwin, O'reilly (SPD) publications.

IV/IV B. TECH. FIRST SEMESTER
INFORMATION SECURITY LAB (Required)

Course Code: CS 7L3 **Credits: 2**
Lecture:-- **Internal assessment: 25 Marks**
Lab: 3period/week **Semester end examination: 50 Marks**

Prerequsite: Information Security

At the end of this course student will:

CO1) Implement the basic cryptographic algorithms to learn how to encrypt and decrypt the messages

CO2) Implement exchange of secret keys without sharing or third party intervention

CO3) Implement digital signatures for the purpose of authentication

CO4) Understand about phishing and find out how to phished popular bank sites in general.

Course Objectives:

1. Practical implementation based on the security applications using JAVA

Syllabus:

1. Write a JAVA program to implement the DES algorithm logic.
2. Write a Java program that contains functions, which accept a key and input text to be

   encrypted/decrypted. This program should use the key to encrypt/decrypt the input by using the

   triple Des algorithm. Make use of Java Cryptography package.
3. Write a JAVA program to implement the Blowfish algorithm logic.
4. Using Java cryptography, encrypt the text “Hello world” using Blowfish. Create your own key

   using Java keytool.
5. Write a Java program to implement RSA algorithm.
6. Implement the Diffie-Hellman Key Exchange mechanism using HTML and JavaScript. Consider

the end user as one of the parties(Alice) and the JavaScript application as the other party(Bob)

7. Calculate the message digest of a text using the SHA-1 algorithm in JAVA.

8. Calculate the message digest of a text using the MD5 algorithm in JAVA.

9. Explore the Java classes related to digital certificates.

10. Create a digital certificate of your own by using the Java key tool.

11. Write a Java program to encrypt users passwords before they are stored in a database table, and to retrieve them whenever they are to be brought back for verification..

12. Write a program in java, which performs a digital signature on a given text.

13. Study phishing in more detail. Find out which popular bank sites have been phished and how.

IV/IV B. TECH. FIRST SEMESTER
MINI PROJECT

Course Code : CS 7L4 **Credits: 2**
Lecture:-- **Internal assessment: 75 Marks**
Lab: 2periods/week **Semester end examination: --**

Course Overview:

This course is intended to implement an industry oriented project by applying the knowledge of Software Engineering, Programming language paradigms, Designing tools, Advanced Technologies and Tools including Testing and Deployment.

Course Outcomes:

At the end of this course student will:

CO1) Apply Process of Project Development to Analyze and design the real world problem.
CO2) Demonstrate the proficiency of Computer Programming Languages & Other Emerging technologies & Tools for Project Implementation.
CO3) Apply various testing techniques and tools to debug the software.
CO4) Deploy the software in the real environment to satisfying the functional and non functional requirements.
CO5) Document the project report of various phases for future scope of the project Development.

IV/IV B. TECH. FIRST SEMESTER
SEMINAR

CS 7L5 **Credits: 1**
Lecture:-- **Internal assessment: 50 Marks**
Lab: 2periods/week **Semester end examination: --**

Course Overview:

This course engages students in the integrated activities of reading, research, discussion, and composition around a designated topic subject. At its core, this course is designed to provide a deeper appreciation of the role of writing in scholarly investigation, as they refine, adapt, and expand their abilities to absorb, synthesize and construct arguments in close.

Course Outcomes:

At the end of this course student will:

CO1) Develop and support a relevant and informed technical topic that is appropriate for its audience, purpose, discipline, and theme.

CO2) Demonstrate effective writing skills and processes by employing the rhetorical techniques of academic writing, including invention, research, critical analysis and evaluation, and revision.

CO3) Effectively incorporate and document appropriate sources in accordance with the formatting style proper for the discipline and effectively utilize the conventions of standard written English.

CO4) Better understand the role that effective presentations have in public/professional contexts and gain experience in formal/informal presentation.

IV/IV B. TECH. SECOND SEMESTER
MANAGERIAL ECONOMICS
AND FI NANCIAL ANALYSIS(Required)

Course Code: CS 8T1 **Credits: 3**
Lecture: 3 periods/ week **Internal assessment: 30 Marks**
Tutorial: 1period/week **Semester end examination: 70 Marks**

Prerequisites: Nil

Course Overview:

Mangers regularly address microeconomic issues ranging from pricing, cost determination, compensation, entry into and exit from markets, and output decisions, Along with that, students should have developed a basic understanding of what financial statements contain and how to use them to assess a company's profitability and financial position. This course introduces various concepts of Economics and Accounting that will be helpful in right decision making.

Course Outcomes:

CO1) Understand of various aspects of managerial economics, demand analysis.

CO2) Understand the use of demand elasticity and various methods of demand forecasting techniques in Managerial decisions.

CO3) Apply production theory, cost theory in relevant Managerial decision making.

CO4) Understand thedifferenced between priceoutput determination under different market conditions along with pricing policies.

CO5) apply capital budgeting, financial analysis techniques in evaluating various investment opportunities.

Syllabus:
Unit -I

Introduction to Managerial Economics & Demand Analysis: Definition of Managerial Economics, Characteristics and Scope – ManagerialEconomics and its relation withother subjects -
Basic economic tools in Managerial Economics. **Demand Analysis:** Meaning- Demand distinctions- Demand determinants- Law of Demand and its exceptions.

Unit-II:**Elasticity of Demand & Demand Forecasting:** Definition -Types of Elasticity of demand - Measurement of price elasticity of demand: Total outlay method , Point method and Arc method= Significance of Elasticity of Demand-Demand Forecasting: Meaning - Factors

governing demand forecasting - Methods of demand forecasting - Criteria of a good forecasting method.

Unit-III

Theory of Production and Cost Analysis: Production Function- Isoquants and Isocosts, MRTS, Law of variable proportions- Law of returns to scale- Least Cost Combination of Inputs, Cobb-Douglas Production function - Economies of Scale. Cost concepts, Opportunity cost, Fixed vs Variable costs, Explicit costs, vs implicit costs, out of pocket costs vs, imputed costs, Determination of Break – Even Point(simple problems) Managerial Significance and limitations of BEP.

Unit –IV Price output determination under different Markets & Pricing Policies: Market Structures: Types of competition, Features of Perfect Competition, Monopoly and Monopoly Competition, Price-Output Determination under Perfect Competition, Monopoly, Monopolistic Competition and Oligopoly Managerial theories of the firm – Marris and Williamson's models. Pricing Policies: Methods of pricing.

Unit –V

Financial Accounting and Capital Budgeting:

Introduction to Double-entry system, Journal, Ledger, Trial Balance – Final Acconts (with simple adjustments) – Limitations of Financial Statements, Ratio Analysis – Liquidity ratios, Profitablity ratios and solvency ratios - **Capital and Capital Budgeting:** Meaning of capital budgeting, Need for capital budgeting – Capital budgeting decisions (Examples of capital budgeting) - Methods of Capital Budgeting: Payback Method, Accounting Rate of Return (ARR), IRR and Net Present Value Method (simple problems)

Text Books:

1. A R Aryasri - Managerial Economics and Financial Analysis, TMH 2011
2. S.A. Siddiqui & A.S. Siddiqui, MangerialEconomice and Financial Analysis, New Age International Publishers, 2011.

Reference Books:

1. N. Appa Rao. & P. Vijaya Kumar - Managerial Economics and Financial Analysis, Cengage Publications, New Delhi, 2011
2. J.V.Prabhakar Rao - Managerial Economics and Financial Analysis, Maruthi Publications, 2011
3. Suma damodaran- Managerial Economics, Oxford 2011

IV/IV B. TECH. SECOND SEMESTER
E-COMMERCE (Elective- III)

Course Code: CS 8T2A **Credits: 3**
Lecture: 3 periods/ week **Internal assessment: 30 Marks**
Tutorial: 1period/week **Semester end examination: 70 Marks**

Prerequisites: Java, Database Management Systems, Advanced Java & Web Technologies

Course Objectives:

1. Mechanism of business transactions through electronic media.
2. Payment transactions in a secured network.
3. Different modes of E-Commerce like Electronic data interchange.
4. Web site establishment, electronic publishing and its importance.

Course Outcomes:

At the end of this course student will:

CO1) Understand the framework and anatomy of ecommerce applications and analyze ecommerce consumer, organizational applications

CO2) Infer mercantile process models from both merchant's and consumer's view point

CO3) Understand the implementation of Electronic Data Interchange (EDI) in day to day life

CO4) Study all the aspects of Intra-Organizational electronic commerce including supply chain management

CO5) Analyze different consumer, information searching methods and resource discovery and information retrieval techniques

Syllabus:

UNIT 1

Electronic Commerce Environment and Opportunities: The Electronic Commerce Environment, Electronic Marketplace Technologies. **Modes of Electronic Commerce:** Electronic Data Interchange, Migration to Open EDI, Electronic Commerce with www/Internet, Commerce Net Advocacy, web Commerce Going Forward.

UNIT 2

Approaches to Safe Electronic Commerce: Secure Transport Protocols, Secure Transactions, Secure Electronic Payment Protocol (SEPP), Secure Electronic Transaction (SET), Certificates for authentication Security on web Servers and Enterprise Networks.

UNIT 3

Electronic Cash and Electronic Payment Schemes: Internet Monetary Payment & Security Requirements. Payment and Purchase Order Process, On-line Electronic cash.

Internet/Intranet Security Issues and Solutions : The need for Computer Security, Specific Intruder Approaches, Security Strategies, Security Tools, Encryption.

UNIT 4

Master Card/Visa Secure Electronic Transaction: Introduction, Business Requirements, Concepts, payment Processing.

E-Mail and Secure E-mail Technologies for Electronic Commerce: The Means of Distribution, A model for Message Handling, Email working, Multipurpose Internet Mail Extensions, Message Object Security Services, Comparisons of Security Methods, MIME and Related Facilities for EDI over the Internet.

UNIT 5

Internet Resources for Commerce: Introduction, Technologies for web Servers, Internet Tools Relevant to Commerce, Internet Applications for Commerce, Internet Charges, Internet Access and Architecture, Searching the Internet.

Advertising on Internet: Issues and Technologies. Advertising on the Web, Marketing creating web site, Electronic

Publishing Issues, Approaches and Technologies: EP and web based EP.

Learning Resource

Text Books

1. Daniel Minoli, Emma Minoli, *Web Commerce Technology Handbook*. TATA

McGraw-Hill Edition.

References

1. Ravi Kalakotar and Andrew B.Whinston, *Frontiers of Electronic Commerce.*
Pearson Education - 1999.

2 Achuyut S.Godbole and Atul Kahate, *Web Technologies TCP/IP to Internet Application Architectures.* Tata McGraw-Hill Publishing Company Limited.

3. Schneider, *Electronic Commerce,* Cengage Publications.

IV/IV B. TECH. SECOND SEMESTER
PARALLEL COMPUTING (Elective- III)

Course Code: CS 8T2B **Credits: 3**
Lecture: 3 periods/ week **Internal assessment: 30 Marks**
Tutorial: 1period/week **Semester end examination: 70 Marks**

Prerequisites: C programming language, Data structures and algorithms.

Course Objectives:

1. To understand the concepts Parallel Computers, Data and Temporal Parallelism.
2. To learn Structures of Parallel Computers.
3. To understand the concepts of Operating Systems for Parallel Computers.
4. To acquire knowledge on CUDA.
5. To learn Parallel Programming with CUDA C.

Course Outcomes:

At the end of this course student will:

CO1) Solve the Problems in Parallel

CO2) Have knowledge on Different Structures of Parallel Computers

CO3) Understand the Performance Evaluation of Parallel Computers

CO4) Get acquaintance on CUDA

CO5) Develop Parallel Programs In CUDA C

Syllabus:

UNIT 1

Inroduction:

Why do we Need High Speed Computing, How do we Increase the Speed of Computers , History of Parallel Computers.

Solving problems in parallel: Utilizing Temporal Parallelism , Utilizing Data Parallelism , Comparison of Temporal and Data Parallel Processing , Data Parallel Processing with Specialized Processors.

UNIT 2

Structure of parallel computers: A Generalized Structure of a Parallel Computer, Classification of Parallel Computers, Vector Computers, A Typical Vector Super Computer, Array Processors, Shared Memory Parallel Computers, Distributed Shared Memory Parallel Computers, Message Passing Parallel Computers.

UNIT 3

Operating systems for parallel computers: Resource Management , Process Management , Process Synchronization , Inter-process Communication , Memory Management , Input/output (Disk Arrays) , Basics of Performance Evaluation , Performance Measurement Tools.

UNIT 4

Computer unified device architecture: The age of parallel processing, The rise of GPU computing, CUDA, Applications of CUDA, Development Environment-CUDA Enabled Graphics Processors, NVIDIA Device driver, CUDA Development Tool kit, Standard C compiler.

UNIT 5

CUDA C:

Introduction to CUDA C: First program, Querying Devices, Using Device Properties,
Parallel Programming in CUDA C: CUDA Parallel Programming- Summing Vectors program

Learning Resource

Text Books

1. Parallel Computers Architecture and Programming, V. Rajaraman, C. Siva Ram Murthy,PHI.
2. CUDA By Example, Jason Sanders, Edward Kandrot, Addison_Wesley.

References

1. Introduction to Parallel Computing, Ananth Grama, Anshul Gupta, George Karypis, Vipin Kumar, Pearson Education.
2. Parallel Computing Theory and Practice, Michel j.Quinn

IV/IV B. TECH. SECOND SEMESTER

SOFTWARE TESTING METHODOLOGIES (Elective- III)

Course Code: CS 8T2C **Credits: 3**

Lecture: 3 periods/ week **Internal assessment: 30 Marks**

Tutorial: 1period/week **Semester end examination: 70 Marks**

Prerequisites:Basic understanding of the software development life cycle (SDLC). basic understanding of software programming using any programming language.

Course Objectives:

1. To study fundamental concepts in software testing
2. To discuss various software testing issues and solutions in software unit test, integration and system testing.
3. To expose the advanced software testing topics, such as object-oriented software testing methods.

Course Learning Outcomes:

At the end of this course student will:

CO1) List a range of different software testing techniques and statergies and be able to apply specific(automated) unit testing method to the projects.

CO2) Distinguish characterstics of structural testing methods.

CO3) Demonstrate the integration testing which aims to uncover interaction and compatibility problems as early as possible.

CO4) Discuss about the functional and system testing methods.

CO5) Demonstrate various issues for object oriented testing.

Syllabus:

Unit-I

A Mathematical Context:
A Perspective on Testing, Examples

Functional Testing: Boundary Value Testing, Equivalence Class Testing, Decision Table-Based Testing, Retrospective on Functional Testing.

Unit-II

Structural Testing:

Path Testing- DD-Paths, Test Coverage Metrics, Basis Path Testing, **Dataflow Testing-** Define/Use Testing, Slice-Based Testing,

Retrospective on Structural Testing- Gaps and Redundancies, Metrics for Method Evaluation.

Unit-III

Integration Testing:

Levels of Testing, Integration Testing- A Closer Look at the SATM System, Decomposition-Based Integration, Call Graph-Based Integration, Path-Based Integration.

Unit – IV

System Testing-

Threads, Basic Concepts for Requirements Specification, Finding Threads, Structural Strategies for Thread Testing, Functional Strategies for Thread Testing SATM Test Threads, System Testing Guidelines

Unit-V

Object-Oriented Testing:

Issues in Object-Oriented Testing, Class Testing, Object-Oriented Integration Testing, GUI Testing, Object-Oriented System Testing.

Learning Resources:

Text Book:

Paul C. Jorgensen, Software Testing: A Craftsman's Approach, 3rd Edition, CRC Press, 2007.

References:

Boris Beizer, Software Testing Techniques, Dreamtech, 2009

IV/IV B. TECH. SECOND SEMESTER
SCRIPTING LANGUAGES (Elective- III)

Course Code: CS 8T2D **Credits: 3**
Lecture: 3 periods/ week **Internal assessment: 30 Marks**
Tutorial: 1period/week **Semester end examination: 70 Marks**

Prerequisites: exposure to C programming and Unix Shell.

Course Objectives:

The course demonstrates an in depth understanding of the tools and the scripting languages necessary for design and development of web applications

Course Outcomes:

At the end of this course student will:

CO1) Identify the differences between typical scripting languages and typical system and application programming languages

CO2) Apply your knowledge of the strengths and weaknesses of scripting languages to select an implementation language

CO3) Create software systems using scripting languages, including PHP and Python

CO4) Write server-side scripts using PHP and Python's CGI facilities

Syllabus:

UNIT 1

PHP Basics PHP Basics- Features, Embedding PHP Code in your Web pages, Outputting the data to the browser, Data types, Variables, Constants, expressions, string interpolation, control structures, Function, Creating a Function, Function Libraries, Arrays, strings and Regular Expressions.

UNIT 2

Advanced PHP Programming Php and Web Forms, Files, PHP Authentication and Methodologies -Hard Coded, File Based, Database Based, IP Based, Login Administration,

UNIT 3

Uploading Files with PHP, Sending Email using PHP, PHP Encryption Functions, the Mcrypt package, Building Web sites for the World – Translating Websites-Updating Web sites Scripts, Creating the Localization Repository, Translating Files, text, Generate Binary Files, Set the desired language within your scripts, Localizing Dates, Numbers and Times.

UNIT 4

Python Introduction to Python language, python-syntax, statements, functions, Built-in-functions and Methods, Modules in python, Exception Handling, Integrated Web Applications in Python

UNIT 5

Building Small, Efficient Python Web Systems ,Web Application Framework.

Learning Resource

Text Books

1. The World of Scripting Languages, David Barron, Wiley Publications.

2. Python Web Programming, Steve Holden and David Beazley, New Riders Publications.
3. Beginning PHP and MySQL, 3rd Edition, Jason Gilmore, Apress Publications (Dreamtech)

References

1. Open Source Web Development with LAMP using Linux, Apache, MySQL, Perl and PHP, J.Lee and B.Ware(Addison Wesley) Pearson Education.

2. Programming Python, M.Lutz,SPD.

3. PHP 6 Fast and Easy Web Development, Julie Meloni and Matt Telles, Cengage Learning Publications.

   a. PHP 5.1, I.Bayross and S.Shah, The X Team, SPD.

   b. Core Python Programming, Chun, Pearson Education.M. TECH. WEB TECHNOLOGIES-R13 Regulations

   c. Guide to Programming with Python, M.Dawson, Cengage Learning.

4. Tcl and the Tk Tool kit, Ousterhout, Pearson Education.

IV/IV B. TECH. SECOND SEMESTER
HUMAN COMPUTER INTERACTION (Elective-IV)

Course Code: CS8T3A **Credits: 3**
Lecture: 3 periods/ week **Internal assessment: 30 Marks**
Tutorial: 1period/week **Semester end examination: 70 Marks**

Prerequisites: Knowledge on software engineering,

Course Objectives: •

1. Design, implement and evaluate effective and usable graphical computer interfaces.
2. Describe and apply core theories, models and methodologies from the field of HCI.
3. Describe and discuss current research in the field of HCI.
4. Describe special considerations in designing user interfaces

Course Outcomes:

At the end of this course student will:

CO1) Understand the concepts and principles of graphical user interface and its design process

CO2) Create effective screen design using screen elements, windows and components

CO3) Select appropriate tool for user interface design

CO4) Identify appropriate user devices for better user interaction

UNIT I:

Introduction: Importance of user interface, definition, importance of good design, Brief history of Screen Design.

Graphical User Interface: Popularity of graphics, the concept of direct manipulation, graphical system, characteristics, Web user – interface popularity, characteristics- principles of user interface.

UNIT II:

Design Process: Human interaction with computers, importance of human characteristics, human considerations, Human interaction speeds.

UNIT III:

Screen designing: Interface design goals, screen meaning and purpose, organizing screen elements, ordering of screen data and content, screen navigation and flow, visually pleasing composition, amount of information, focus and emphasis, presenting information simply and meaningfully, technological considerations in interface design.

UNIT IV:

Windows: Characteristics, components, operations. Selection of device based and screen based controls.

UNIT – V:

Components: Icons and images, Multimedia, choosing proper colors

Interaction devices: Keyboard and function keys, pointing devices, speech recognition, digitization and generation, image and video displays, drivers

Learning Resources:

Text Books:

1. Wilbert O Galitz, The Essential Guide to User Interface Design. 2 ed, Wiley DreamaTech
2. Ben Shneidermann, Designing the User Interface. 3 ed, Pearson Education Asia

Reference Books:

1. Alan Dix, Janet Fincay, Gre Goryd, Abowd and Russell Bealg, Human Computer Interaction. Pearson.
2. Prece, Rogers, Sharps Interaction Design. Wiley Dreamatech,
3. Soren Lauesen, User Interface Design. Pearson Education.

E-learning resources:

http://onlinevideolecture.com/?course_id=1329&lecture_no=9
https://class.coursera.org/hci/lecture

IV/IV B.TECH. SECOND SEMESTER

TCP/IP (Elective- IV)

Course Code: CS8T3B **Credits: 3**
Lecture: 3 periods/week **Internal assessment: 30 marks**
Tutorial: 1 period /week **Semester end examination: 70 marks**

Prerequisite : Computer Networks

Course Objectives :

1. This course provides a solid foundation for understanding the communication process of the Internet.
2. The student will understand the fundamental concepts of computer networking in the context of the TCP/IP model and protocols.
3. To study classful and classless addressing, IPV4,IPv6, UDP, TCP, congestion control and flow control.

Course Outcomes :

At the end of this course student will:

CO1) Summarize basic principles of IPv4 and its Addressing mechanisms

CO2) Understand UDP Services and Applications in Transport Layer

CO3) Describe the services, and features of TCP

CO4) Discuss various Flow , Error and Congestion control mechanisms of TCP

CO5) Understand the Principles of IPv6 Addressing ,IPv6 and ICMPv6 Protocols

Syllabus

UNIT – I

The OSI Model and the TCP/IP Protocol Suite - Protocol Layers, The OSI Model, TCP/IP Protocol suite and Addressing.**IPV4 Addresses**- Introduction, Classful and Classless Addressing, **Internet Protocol Version4(IPv4)** – Datagrams, Fragmentation, Options, Checksum, Security, IP Package.

UNIT- II

Introduction to the Transport Layer – Transport Layer Services and Protocols. **User Datagram Protocol(UDP)** – Introduction, User Datagram, UDP Services and Applications, UDP Package.

UNIT – III

Transmission Control Protocol – I : TCP Services, Features, Segment, TCP Connection, Windows in TCP.

UNIT – IV

Transmission Control Protocol – II : Flow Control, Error Control, Congestion Control, TCP Timers, Options and TCP Package.

UNIT – V

IPv6 Addressing – Introduction, Address Space Allocation, Global Unicast Addresses, Autoconfiguration and Renumbering. **IPv6 Protocol -** Introduction , Packet Format, Transition from IPv4 to IPv6. **ICMPv6 -** Introduction, Error Messages, Informational Messages, Neighbor-Discovery Messages, Group Membership Messages.

Learning Resources

Text Book :

1. TCP/IP Protocol Suite , Behrouz A. Forouzan, 4th Edition, Tata McGraw-Hill Edition.

IV/IV B. TECH. SECOND SEMESTER
VIRTUAL REALITY (Elective- IV)

Course Code: CS8T3C **Credits: 3**
Lecture: 3 periods/ week **Internal assessment: 30 Marks**
Tutorial: 1period/week **Semester end examination: 70 Marks**

Prerequisite: Data Structures

Course Objectives :

1. Virtual reality in different object & applications.
2. Virtualization of image having big data.
3. High performance of computing with virtual reality.

Course Outcomes:

At the end of this course student will:

CO1) Understand the components of the virtual reality system

CO2) Describe various input and output devices used for virtual reality

CO3) Apply the different modelling concepts to visual virtualization

CO4) Analyze the performance of given simple applications related to virtual reality

CO5) Implement 3D technology with virtual programming concepts

Syllabus

UNIT-I

Introduction : The three I's of virtual reality, commercial VR technology and the five classic components of a VR system. (1.1, 1.3 and 1.5 of Text Book (1))

UNIT – II

Input Devices : (Trackers, Navigation, and Gesture Interfaces): Three-dimensional position trackers, navigation and manipulation, interfaces and gesture interfaces. (2.1, 2.2 and 2.3 of Text Book (1)).

Output Devices: Graphics displays, sound displays & haptic feedback. (3.1,3.2 & 3.3 of Text Book (1))

UNIT – III

Modeling : Geometric modeling, kinematics modeling, physical modeling, behaviour modeling, model management. (5.1, 5.2 and 5.3, 5.4 and 5.5 of Text Book (1)).

UNIT – IV

Human Factors: Methodology and terminology, user performance studies, VR health and safety issues. (7.1, 7.2 and 7.3 of Text Book (1)).

Applications: Medical applications, military applications, robotics applications. (8.1, 8.3 and 9.2 of Text Book (1)).

UNIT – V

VR Programming-I : Introducing Java 3D, loading and manipulating external models, using a lathe to make shapes. (Chapters 14, 16 and 17 of Text Book (2)) **VR Programming-II :** 3D Sprites, animated 3D sprites, particle systems. (Chapters 18, 19 and 21 of Text Book (2))

Learning Resources

TEXT BOOKS :

1. Virtual Reality Technology, Second Edition, Gregory C. Burdea & Philippe Coiffet, John Wiley & Sons, Inc.,
2. Killer Game Programming in Java, Andrew Davison, Oreilly-SPD, 2005.

REFERENCES :

1. Understanding Virtual Reality, interface, Application and Design, William R.Sherman, Alan Craig, Elsevier(Morgan Kaufmann).
2. 3D Modeling and surfacing, Bill Fleming, Elsevier(Morgan Kauffman).
3. 3D Game Engine Design, David H.Eberly, Elsevier.
4. Virtual Reality Systems, John Vince, Pearson Education.
5. What is Virtual Reality?, http://vr.isdale.com/WhatIsVR/frames/WhatIsVR4.1.html.
6. Augumented and Mixed Reality, http://www.mic.atr.co.jp/~poup/research/ar/.

IV/IV B. TECH. SECOND SEMESTER
SECURE DATABASE APPLICATION DEVLOPMENT (Elective- IV)

Course Code: CS 8T3D **Credits: 3**
Lecture: 3 periods/ week **Internal assessment: 30 Marks**
Tutorial: 1period/week **Semester end examination: 70 Marks**

Prerequisite : Database Management Systems, Information Security

Course Objectives:

1. This course aims to give an overview of various types of security threats to database and operating System. This course also covers the aspects of user administration, setting of user profiles, password policies, privileges and roles to users. This course also covers the various database security models, and database auditing models. This course will give the practical implementation of the above in Oracle and Microsoft SQL Server.

Course Outcomes:

At the end of this course student will:

CO1) Understand the need for security for the database and operating systems

CO2) Create users, and apply user Profiles, Password Policies, Privileges, and Roles

CO3) Understand various Database Application Security Models and their advantages

CO4) Understand the need for database auditing

CO5) Apply necessary auditing policies for database

Syllabus:

UNIT 1

Security Architecture: Introduction, Security, Information Systems, Database management systems, Information security, Information security Architecture, database security, Asset types and their value, Security methods.

Operating System Security Fundamentals: Introduction, operating systems overview, security environment, components, Authentication methods, user administration, password policies, Vulnerabilities of operating systems, E- Mail security.

UNIT 2

Administration of Users: Introduction, user authentication, operating system authentication, creating/removing/modifying users, default/remote users, Database links, Linked servers, remote servers.

Profiles, Password Policies, Privileges, and Roles: Introduction, Defining and using profiles, Designing and implementing password policies, Granting and revoking user privileges, creating, Assigning and revoking user roles.

UNIT 3

Database Application Security Models: Introduction, Types of users, security models, application types, application security models and Data encryption. Virtual Private Databases (VPD): Introduction, Overview, implementing a VPD using views and application context. Implementing oracle VPD, Viewing VPD policies and application context using: data dictionary, policy manager, implementing row and column level security with SQL server.

UNIT 4

Database Auditing Models, Application Data Auditing: Database Auditing Models: Introduction, Auditing overview, environment, process, objectives, classification and types, benefits and side effects of auditing.

Application Data Auditing: Introduction, DML auction auditing architecture. Triggers, fine grained auditing, DML statement audit trail and auditing application errors with Oracle.

UNIT 5

Auditing Database Activities, Security and Auditing Project Cases: Auditing Database Activities: Introduction, usage of database activities, creating DLL triggers, auditing database activities with oracle.

Security and Auditing project cases: Introduction, case study for developing an online database, taking care of payroll, tracking database changes and developing a secured authentication repository.

Learning Resource

Text Books

1. Database Security and Auditing, Hassan Afyouni, Cengage Learning, 2007
2. Database Security, S. Castano, M. Fugini, G. Martella, P. Samarati, Addison-Wesley, 1994
3. Implementing Database Security and Auditing, RonBen Natan: Elsevier, Indian reprint, 2006

References

1. Principles of Distributed Database Systems, Prentice Hall,2/e, M.TamerÖzsu, Patrick Valdureiz
2. Database Security, Castano, Fugini, Addison Wesley
3. The security Audit and control of Databases, Clark, Holloway, List, UK:Ashgate.
4. Security and Audit of Database System, Douglas, Blackwell(UK)
5. Database security and Integrity, Fernandez, Summers, Wood, Addison Wesley

IV/IV B. TECH. SECOND SEMESTER
MAJOR PROJECT

Course Code : CS 8PW **Credits: 9**
Lab: 9 periods/ week **Internal assessment: 100Marks**
Tutorial: -- **Semester end examination:200Marks**

Course Overview:

This course is intended to implement an industry oriented project by applying the knowledge of Software Engineering, Programming language paradigms, Designing tools, Advanced Technologies and Tools including Testing and Deployment.

Course Outcomes:

At the end of this course student will:

CO1) Apply Process of Project Development to Analyze and design the real world problem.
CO2) Demonstrate the proficiency of Computer Programming Languages & Other Emerging technologies & Tools for Project Implementation.
CO3) Apply various testing techniques and tools to debug the software.
CO4) Deploy the software in the real environment to satisfying the functional and non functional requirements.
CO5) Document the project report of various phases for future scope of the project Development.