K.R. MANGALAM UNIVERSITY

THE COMPLETE WORLD OF EDUCATION

SCHOOL OF ENGINEERING AND TECHNOLOGY

Bachelor of Science (Honours) Computer Science

B.Sc (H) (CS)

Programme Code:72

2021-24

Approved in the 26th Meeting of Academic Council Held on 11 August 2021

Member Secretary
Academic Council
K.R. Mangalam University

Registrar
K.R. Mangalam University
Sohna Road, Gurugram, (Haryana)

K.R. MANGALAM UNIVERSITY

THE COMPLETE WORLD OF EDUCATION

SCHOOL OF ENGINEERING

AND

TECHNOLOGY

Bachelor of Science (Honours) Computer Science

B.Sc (H) (CS)

Programme Code:72

2021-24

Approved in the 26th Meeting of Academic Council Held on 11 August 2021

PREFACE

Computer science is a discipline that spans theory and practice and it requires thinking both in abstract terms and in concrete terms. In last one decade the discipline of computer science has made huge strides. Many problems in science, engineering, health care, business, and other areas can be solved effectively with computers, but finding a solution requires both computer science expertise and knowledge of the particular application domain. "Data is new oil" credited to Mathematician Clive Humby, has gained more significance with emerging technologies like Artificial Intelligence, Machine Learning and Data Science.

In consultation with Deans, Faculty Members, Industry Experts, and University Alumni, the Academic council constituted school-wise committees to draft the curriculum of B.Sc. (H) Computer Science. The primary emphasis is to designing a course that combines courses from the disciplines of Statistics, Mathematics, and Computer Science and prepares students for careers in Big Data Science & Analytics

The B.Sc.(H) Computer Science program is spread over three years in six semesters. The total numbers of credits are 148. The program is designed as per LOCF guidelines laid by UGC and run in collaboration with IBM. The core courses includes specialized courses like clean coding with Python, Big Data Analytics and Data Warehouse and Data Mining. The courses contents and electives are aims at laying a strong foundation of CS at an early stage of the career along with two other subjects such as Physics, Mathematics, Electronics, Statistics etc. The Discipline Specific elective addresses the need to familiarize students with emerging areas in computer science. The laboratories, besides supplementing the theory course should also expose the student to the use of the latest software tools.

The present curriculum focuses on unique interdisciplinary educational experience allows students the opportunity to acquire the broad base of knowledge and skills which employers are seeking. The programme is designed to attract international students making K.R. Mangalam a global place of higher learning and research in engineering and technology.

About K.R Mangalam University

The K.R. Mangalam Group has made a name for itself in the field of education. The K.R. Mangalam story goes back to the chain of schools that offered an alternative option of world-class education, pitching itself against the established elite schools, which had enjoyed a position of monopoly till then. Having blazed a new trail in school education, the focus of the group was aimed at higher education.

K.R. Mangalam University is the fastest-growing higher education institute in Gurugram, India. K. R. Mangalam University was established under the Haryana Private University Act 2006, received the approval of Haryana Legislature vide Amendment Act # 36 of 2013 and consent of the Hon'ble Governor of Haryana on 11th April 2013, which was published in the Gazette notification vide Leg. No.10/2013, dated 3rd May 2013.

Since its inception in 2013, the University has been striving to fulfil its prime objective of transforming young lives through ground-breaking pedagogy, global collaborations, and world-class infrastructure. Resources at K.R Mangalam University have been continuously upgraded to optimize opportunities for the students. Our students are groomed in a truly interdisciplinary environment where they grow up with integrative skills through interaction with students from engineering, social sciences, management and other study streams.

K.R Mangalam University is unique because of its:

i. Enduring legacy of providing education to high achievers who demonstrate leadership in diverse fields.

ii. Protective and nurturing environment for teaching, research, creativity, scholarship, social and economic justice.

Objectives

i. To impart undergraduate, post graduate and doctoral education in identified areas of higher education.

ii. To undertake research programmes with industrial interface.

iii. To integrate its growth with the global needs and expectations of the major stake holders through teaching, research, exchange & collaborative programmes with foreign, Indian Universities/Institutions and MNCs.

iv. To act as a nodal center for transfer of technology to the industry.

v. To provide job oriented professional education to the Indian student community with particular focus on Haryana.

About School of Engineering & Technology (SOET)

School of Engineering and Technology (SOET), K.R. Mangalam University is dedicated to fostering innovation, excellence, and advancement in engineering and technology. Empowering the new generation of change-makers by imparting exceptional understanding and intellect to facilitate the creation of highly sophisticated futuristic solutions. Our well-qualified academicians, accomplished researchers and industry insiders are focused on imparting their extensive knowledge and expertise to students through various lectures, workshops, industrial visits, projects, and competitions throughout the year ensuring that students receive a comprehensive education that blends theory with practical application.

These programs offered at SOET have the distinct objective of equipping the students with knowledge, skills and attitudes in engineering and technology, to make them capable of successfully meeting the present requirements and future challenges in the engineering profession. SOET brings together outstanding academics, industry professionals, and experienced researchers to deliver a unique hands-on and multi-disciplinary learning experience.

The curriculum of programs has been designed to cater to the ever changing needs and demands of the industry. The curriculum is regularly updated. The school has best infrastructure including domain-specific labs. SOET aims to provide exposure to the principles and practices of Design / Developments and Projects in the area of engineering. SOET is offering Ph.D. programs also.

School Vision

To create, disseminate, and apply knowledge in science and technology to meet the higher education needs of India and the global society, To serve as an institutional model of excellence in scientific and technical education characterized by integration of teaching, research and innovation.

School Mission

M1: To create an environment where teaching and learning are prioritized, with all support activities being held accountable for their success.

M2: To strengthen the institution's position as the school of choice for students across the State & Nation.

M3: To promote creative, immersive, and lifelong learning skills while addressing societal concerns.

M4: To promote co- and extra-curricular activities for overall personality development of the students.

M5: To promote and undertake all-inclusive research and development activities.

M6: To instill in learners an entrepreneurial mindset and principles.

M7: Enhance industrial, institutional, national, and international partnerships for symbiotic relationships.

M8: To help students acquire and develop knowledge, skills and leadership qualities of the 21st Century and beyond.

Programmes offered by the School

School offers undergraduate B. Tech Program, B.Sc. (Hons) Program, postgraduate M. Tech Program, and Doctoral Program. All these programs are designed to impart scientific knowledge to the students and provide theoretical and practical training in their respective fields.

B.Sc.(H) Computer Science

This program is designed to provide a sound knowledge of computing principles and applications in scientific and engineering domains. It develops the ability to analyze problems and generate solutions in the areas of computing. It also aims to provide exposure to the principles and practices of design and development of computing system related to Learning Outcome Based Curriculum Framework (LOCF) and Choice Based Credit System (CBCS).

Eligibility Criteria: The student should have passed the 10+2 examination conducted by the Central Board of Secondary Education or equivalent examination from a recognized Board with mathematics as one of the subjects and with an overall aggregate of 50% or more.

Course Outline: Python Programming / Operating Systems/ Computer networks / Mobile Application Development / Databases / Big Data Analytics / Artificial Intelligence.

Career Options:- Opportunities exist in IT industry, freelancers, Data Scientist, AI/ ML Personnel.

Program Educational Objectives (PEO)

PEO 1: To develop graduates who have strong foundation of knowledge and skills in the field of computer science and engineering.

PEO 2: To develop graduates who are employable in industries/public sector/research organizations or work as an entrepreneur.

PEO 3: To foster graduates who can provide solutions to challenging problems in their profession by applying computer engineering theory and practices.

PEO 4: To encourage graduates who can provide leadership and are effective in multidisciplinary environment.

PEO 5: To develop ability to demonstrate team work with the ability of leadership, analytical reasoning for solving time critical problems and strong human values for responsible professional.

PEO 6: To impart knowledge and skills to analyze, design, test and implement diverse range of technology.

Program Outcomes (PO)

PO 1 Engineering Knowledge: Apply knowledge of mathematics, science, engineering fundamentals and an engineering specialization to the solution of complex engineering problems.

PO 2 Problem analysis: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO 3 Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

PO 4 Conduct investigations of complex problems: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO 5 Modern tool usage: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

PO 6 The engineer and society: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering employability.

PO 7 Environment and sustainability: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

PO 8 Ethics: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO 9 Individual and team work: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

PO 10 Communication: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO 11 Project management and finance: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects through entrepreneurship skills and in multidisciplinary environments.

PO 12 Life-long learning: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change through skill development.

Program Specific Outcome: B.Sc.(H) Computer Science

PSO 1 Employability: Demonstrate mastery of computer science in the core knowledge areas to take up technical employment and cultivate skills for thriving career and higher studies
PSO 2 Ethics and Behaviour: Acquaint with the contemporary trends in industrial/ research settings and display ethical code of conduct in usage of Internet and Cyber systems.
PSO 3 Knowledge: Address one's disciplinary knowledge and transferable skills to new/unfamiliar contexts and to identify and analyze problems and issues and seek solution to real-life problems.

Program Duration

The maximum completion period of the B.Sc. (H) Computer Science program offered by the University shall be three years

Class Timings

The classes will be held from Monday to Friday from 09:10 am to 04:00 pm.

Scheme of Studies and Syllabi

The scheme of studies and syllabi of B.Sc. (H) Computer Science program for all semesters is given in the following pages. These are arranged as semester-wise.

For each course, the first line contains Course Code and Credits (C) of the course.
This is followed by the course objectives, course outcome and the syllabus (Unit I to IV), Text book and reference books.

Three Years B.Sc. (H) Computer Science Program at a Glance

	Semester I	Semester II	Semester III	Semester IV	Semester V	Semester VI	Total
Course	8	7	11	8	9	9	52
Credit	23	22	29	25	20	29	148

Scheme of Studies as per Learning Outcome Based Curriculum Framework (LOCF) and Choice Based Credit System (CBCS)

SEMESTER I

ODD SEMESTER								
SNO	**Category**	**COURSE CODE**	**COURSE TITLE**	**L**	**T**	**P**	**C**	**EMP/ENT/SE/OP**
1	SEC	ETCS316A	Web Technologies	3	1	-	3	SE/EMP/OP
2	GE	BSPH103A	MECHANICS	4		-	4	SE
3	AEC	UCCS 155A	COMMUNICATION SKILLS	4	-	-	4	SE
4	CC	ETCS106A	Clean Coding with Python	3	0	0	3	EMP
5	AEC	UCDM301A	Disaster Management	3	-	-	3	SE
6	SEC	ETCA164A	WEB TECHNOLOGIES LAB	-	-	2	1	SE/EMP/OP
7	CC	ETCS157A	Clean Coding with Python Lab	-	-	2	1	EMP
8	GE	BSPH153A	MECHANICS Lab	0	0	4	2	GE
			TOTAL	**17**	**1**	**8**	**21**	

SEMESTER II

EVEN SEMESTER							
SNO	**Category**	**COURSE CODE**	**COURSE TITLE**	**L**	**T**	**P**	**C**
1	GE	ETPH102A	ELECTRICITY AND MAGNETISM	4		-	4
2	AEC	UCES125A	ENVIRONMENTAL STUDIES	3	-	-	3
3	CC	ETCS307A	DATABASE MANAGEMENT SYSTEMS	3	1	-	4
4	SEC		Open Elective	3	-	-	3
5	CC	ETCS112A	OBJECT ORIENTED PROGRAMMING	3	1	-	4
6	CC	ETCS 355A	DATABASE MANAGEMENT SYSTEMS LAB	-	-	2	1
7	CC	ETCS166A	OBJECT ORIENTED PROGRAMMING LAB	-	-	2	1
8	GE	ETPH152A	ELECTRICITY AND MAGNETISM Lab	-	-	4	2
			TOTAL	**16**	**2**	**8**	**22**

SEMESTER III

1	SEC	ETCS321A	Java Programming	3	1	-	4	SE/EMP/OP
2	CC	ETCS211A	OPERATING SYSTEMS	3	1	-	4	SE
3	GE	ETEC 210A	DIGITAL ELECTRONICS	3	1	-	4	SE
4	PCC	ETCS231A	Discrete Mathematics	3	1	-	4	SE/OP
5	CC	ETCS217A	DATA STRUCTURES	3	1	-	4	SE/EMP
6	GE	ETMA215A	PROBABILITY AND STATISTICS	4	-	-	4	SE
7	CC	ETCS 257A	DATA STRUCTURES LAB	-	-	2	1	SE/EMP

8	CC	ETCS255A	OPERATING SYSTEMS LAB	-	-	2	1	SE
9	SEC	ETCS367A	Java Programming Lab	-	-	2	1	SE/EMP
10	GE	ETEC 256A	DIGITAL ELECTRONICS LAB	-	-	2	1	SE
			TOTAL	**23**	**5**	**10**	**28**	

SEMESTER IV

1	CC	ETCS222A	COMPUTER ORGANIZATION & ARCHITECTURE	3	1	-	4	SE
2	CC	ETCS 220A	ANALYSIS AND DESIGN OF ALGORITHMS	3	1	-	4	SE/EMP
3	SEC	ETCA365A	Linux Environment Lab	-	-	2	1	SE
4	CC	ETCS202A	SOFTWARE ENGINEERING	3	1	-	4	EMP/ENT/OP
5	SEC	ETCA228A	Mobile Application Development	4	-	-	4	EMP/ENT
6	CC	ETCS260A	Computer Organization & Architecture Lab	-	-	2	1	SE
7	CC	ETCS262A	Analysis and Design of Algorithms Lab	-	-	2	1	SE/EMP
8	CC	ETCS252A	SOFTWARE ENGINEERING LAB	-	-	2	1	EMP/ENT/OP
9	SEC	ETCA264A	Mobile Application Development Lab	-	-	2	1	EMP/ENT
			TOTAL	**16**	**3**	**10**	**21**	

SEMESTER V

1	CC	ETCS 214A	Theory of Computation	3	1	-	4	SE
2	CC	ETCS304 A	COMPUTER NETWORKS	3	1	-	4	EMP/OP
3	CC	ETCS 206A	COMPUTER GRAPHICS	3	1	-	4	SE
4	CC	ETCS365 A	COMPUTER NETWORKS LAB	-	-	2	1	EMP/OP
5	CC	ETCS258 A	COMPUTER GRAPHICS LAB	-	-	2	1	SE
6	CC	ETCS308 A	Big Data Analytics	3	-	-	3	EMP/ENT
7	CC	ETCS364 A	Big Data Analytics Lab	-	-	2	1	EMP/ENT
8	DSE		DSE -1	3	-	-	3	
9	DSE		DSE -1 LAB	-	-	2	1	
10	DSE		DSE - 2	3	-	-	3	
11	DSE		DSE - 2 LAB	-	-	2	1	
			TOTAL	**21**	**3**	**10**	**26**	

DSE -1						
(i)	ETCS409A	Advanced Computer Networks	3	-	-	3
	ETCS452A	Advanced Computer Networks Lab	-	-	2	1
(ii)	ETCS410A	Mobile and Wireless Communication	3	-	-	3
	ETCS453A	Mobile and Wireless Communication Lab	-	-	2	1
(iii)	ETCS411A	Machine Learning	3	-	-	3
	ETCS455A	Machine Learning Lab	-	-	2	1

DSE -2							
(i)	ETCS517A	Soft Computing	3	-	-	3
	ETCS559A	Soft Computing Lab	-	-	2	1
(ii)	ETCS519A	Big Data Analytics and Visualization	3	-	-	3
	ETCS563A	Big Data Analytics and Visualization	-	-	2	1
(iii)	ETCS515A	Ethical Hacking	3	-	-	3
	ETCS557A	Ethical Hacking Lab	-	-	2	1

SEMESTER VI

1	SEC	ETCA324A	.Net FRAMEWORK	3	1	-	4	EMP/ENT
2	CC	ETCS520A	Internet Technologies	3	1	-	4	EMP
3	CC	ETCS401A	Artificial Intelligence	3	1	-	4	EMP/ENT
4	CC	ETCS451A	Artificial Intelligence Lab	-	-	2	1	EMP/ENT
5	SEC	ETCA364A	.Net FRAMEWORK LAB	-	-	2	1	EMP/ENT
6	CC	ETCS519A	BLOCKCHAINS	3	1	-	4	EMP/ENT
7	DSE		DSE -3	4	-	-	4	
8	DSE		DSE -3 LAB	-	-	2	1	
9	PROJ	ETCS464A	Major Project	-	-	-	6	EMP/ENT
			TOTAL	**16**	**4**	**6**	**29**	
			TOTAL CREDITS				**147**	

DSE -3						
(i)	ETCS422A	Cloud Computing	4	-	-	4
	ETCA362A	Cloud Computing Lab	-	-	2	1
(ii)	ETCS424A	Data Warehousing and Data Mining	4	-	-	4
	ETCS463A	Data Warehousing and Data Mining Lab	-	-	2	1
(iii)	ETCS421A	Internet of Things	4	-	-	4
	ETCS457A	Internet of Things Lab	-	-	2	1

Semester I

ETCS316A	Web Technologies	L	T	P	C
Version 1.0		3	1	0	4
Pre-requisites/Exposure	Basics of HTML				
Co-requisites	--				

Course Objectives

1. Analyze a web page and identify its elements and attributes.
2. Create web pages using XHTML and Cascading Style Sheets.
3. Build dynamic web pages using JavaScript (Client side programming).
4. Create XML documents and Schemas.
5. Build interactive web applications using AJAX.

Course Outcomes

Upon completion of the course the students will be able to:

CO1. Create a well-designed and well-formed, professional Web site utilizing the most current standards and practices
CO2. Demonstrate knowledge in web technologies including HTML, XHTML, CSS, image editing software, web authoring software, and client-side scripting
CO3. Create client-side scripts to add interactivity to Web pages
CO4. Select appropriate Web tools for a Web development project
CO5. Identify Web authoring obstacles created by the availability of various web browsers and markup language versions

Catalog Description

This course is an introduction to Web site development and the technologies behind it. Students will learn how to design and develop Web pages using current technologies and tools. Topics covered will include the World Wide Web, HTML, Cascading Style Sheets (CSS) and XML. The focus of this course is on dynamic HTML, a collection of web technologies such as HTML and scripting languages used together to create interactive and animated Web pages. Students will learn to program client-side scripts using JavaScript and the Document Object Model to transform static Web pages created with HTML and CSS into dynamic Web pages.

Course Content

Unit I: **8 lecture hours**

Concept of WWW, Internet and WWW, HTTP Protocol: Request and Response, Web browser and Web servers, Features of Web 2.0, Common terminology: IP Addressing, URLs, Domain names. Website Creation and maintenance, Web Hosting and Publishing Concepts, Search Engines and their working. HTML: Introduction to HTML, HTML Document structure tags, HTML comments, Text formatting, inserting special characters, anchor tag, adding images and sound, lists: types of lists, tables, frames and floating frames, Developing Forms, Image maps, formatting and fonts, commenting code, color, hyperlink, lists, tables, images, forms, XHTML, Meta tags, Character entities, frames and frame sets,

Unit II: 12 lecture hours

Client-side scripting: JavaScript - Data Types, Control Statements, operators, Built-in and User Defined Functions, Objects in JavaScript, Handling Events. HTML Document Object Model. Page Styling: Separation of content and presentation in HTML, Cascading Style Sheets - Types of Style Sheets – Internal, inline and External style sheets, customizing common HTML elements, types of CSS selectors

Unit III: 12 lecture hours

Concepts of effective web design, Web design issues including Browser, Bandwidth and Cache, Display resolution, Look and Feel of the Website, Page Layout and linking, User centric design, Sitemap, Planning and publishing website, Designing effective navigation, Browser architecture and Web site structure

Unit IV: 8 lecture hours

XML: Introduction to XML-Mark up languages, Features of Mark-up languages, XML Naming rules, Building block of XML, Document, Difference between HTML & XML, Components of XML, XML Parser, DTD's Using XML with HTML and CSS.
Introduction to Web Services, UDDI, SOAP, WSDL, Web Service Architecture, Developing and deploying web services. AJAX –Introduction AJAX programming, Improving web page performance using AJAX.

Text Books

1. Internet and World Wide Web, Deitel H.M., P.J.Deitel , Pearson

Reference Books/Materials

1.Web Technologies, Uttam K. Roy, Oxford University Press
2. HTML Black Book, Stephen Holzner, Wiley Dreamtech.
3. Web Technology, Rajkamal, Tata McGraw-Hill.
4. Web Technologies: A Computer Science Perspective, Jeffrey C. Jackson, Pearson.
5. XML: How to Program, Deitel&Deitel Nieto

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination
Examination Scheme:

Components	Quiz	Attendance	Mid Term Exam	Presentation/ Assignment/ etc.	End Term Exam
Weightage (%)	10	10	20	10	50

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Create a well-designed and well-formed, professional Web site utilizing the most current standards and practices	**PO1**
CO2	Demonstrate knowledge in web technologies including HTML, XHTML, CSS, image editing software, web authoring software, and client-side scripting	**PO4**
CO3	Create client-side scripts to add interactivity to Web pages	**PO5**
CO4	Select appropriate Web tools for a Web development project	**PO2**
CO5	Identify Web authoring obstacles created by the availability of various web browsers and markup language versions	**PO3**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2	PSO3
ETCS316A	WEB TECHNOLOGIES	2	2	2	3	3										3

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETPH103A	Mechanics	L	T	P	C
Version 1.0		4	0	0	4
Pre-requisites/Exposure	Basic Physics and Mathematics				
Co-requisites					

Course Objectives

1. To acquire the knowledge of fundamentals of motion of objects, work, energy and collisions

2. To understand the concepts of rotational dynamics, elasticity and fluid motion.

3. To gain insight to the theory of gravitation and oscillations.

4. To have an insight about non-inertial systems and Special Theory of Relativity.

Course Outcomes

On completion of this course, the students will be able to

CO1. Better understand the laws of physics governing the motion of physical objects and relationship between force, work and energy.
CO2. Comprehend the concept of rotational motion of objects, elastic properties of the materials and motion of fluids.
CO3. Have an understanding of motion under gravitational force of attraction and simple harmonic motion.
CO4. Gain deeper understanding of Special Theory of Relativity, Lorentz Transformation, Mass energy transformations.

Catalog Description

This course is intended to introduce the basic concepts of physics related to motion of objects. It discusses the logic behind many events that we observe around us in day to day life. It emphasizes on the relationship between force, matter and motion under different circumstances. This course is a bridge between the basic and advance concepts of physics.

Course Content

Fundamentals of Dynamics: Reference frames. Inertial frames; Review of Newton's Laws of Motion. Galilean transformations; Galilean invariance. Momentum of variable-mass system: motion of rocket. Motion of a projectile in Uniform gravitational field Dynamics of a system of particles. Centre of Mass. Principle of conservation of momentum. Impulse. (6 Lectures)

Work and Energy: Work and Kinetic Energy Theorem. Conservative and non-conservative forces. Potential Energy. Energy diagram. Stable and unstable equilibrium. Elastic potential energy. Force as gradient of potential energy. Work & Potential energy. Work done by non conservative forces. Law of conservation of Energy. (4 Lectures)

Collisions: Elastic and inelastic collisions between particles. Centre of Mass and Laboratory frames. (3 Lectures)

Rotational Dynamics: Angular momentum of a particle and system of particles. Torque. Principle of conservation of angular momentum. Rotation about a fixed axis. Moment of Inertia. Calculation of moment of inertia for rectangular, cylindrical and spherical bodies. Kinetic energy of rotation. Motion involving both translation and rotation. (12 Lectures)

Elasticity: Relation between Elastic constants. Twisting torque on a Cylinder or Wire.(3 Lectures)

Fluid Motion: Kinematics of Moving Fluids: Poiseuille's Equation for Flow of a Liquid through a Capillary Tube. (2 Lectures)

Gravitation and Central Force Motion: Law of gravitation. Gravitational potential energy. Inertial and gravitational mass. Potential and field due to spherical shell and solid sphere. (3 Lectures)
Motion of a particle under a central force field. Two-body problem and its reduction to one-body problem and its solution. The energy equation and energy diagram. Kepler's Laws. Satellite incircular orbit and applications. Geosynchronous orbits. Weightlessness. Basic idea of globalpositioning system (GPS). (6 Lectures)

Oscillations: SHM: Simple Harmonic Oscillations. Differential equation of SHM and its solution. Kinetic energy, potential energy, total energy and their time-average values. Dampedoscillation. Forced oscillations: Transient and steady states; Resonance, sharpness of resonance; power dissipation and Quality Factor. (7 Lectures)

Non-Inertial Systems: Non-inertial frames and fictitious forces. Uniformly rotating frame. Laws of Physics in rotating coordinate systems. Centrifugal force. Coriolis force and its applications. Components of Velocity and Acceleration in Cylindrical and Spherical Coordinate Systems. (4 Lectures)

Special Theory of Relativity: Michelson-Morley Experiment and its outcome. Postulates of Special Theory of Relativity. Lorentz Transformations. Simultaneity and order of events. Lorentz contraction. Time dilation. Relativistic transformation of velocity, frequency and wave number.

Relativistic addition of velocities. Variation of mass with velocity. Massless Particles. Mass energy Equivalence. Relativistic Doppler effect. Relativistic Kinematics. Transformation of Energy and Momentum. (10 Lectures)

Text books:

1. Mechanics, D.S. Mathur, S. Chand and Company Limited, 2000

Reference book(s) [RB]:

1. An introduction to mechanics, D. Kleppner, R.J. Kolenkow, 1973, McGraw-Hill.
2. Mechanics, Berkeley Physics, vol.1, C.Kittel, W.Knight, et.al. 2007, Tata McGraw-Hill.
3. Physics, Resnick, Halliday and Walker 8/e. 2008, Wiley.
4. Analytical Mechanics, G.R. Fowles and G.L. Cassiday. 2005, Cengage Learning
5. Feynman Lectures, Vol. I, R.P.Feynman, R.B.Leighton, M.Sands, 2008, Pearson Education
6. Introduction to Special Relativity, R. Resnick, 2005, John Wiley and Sons.
7. University Physics, Ronald Lane Reese, 2003, Thomson Brooks/Cole.

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination
Examination Scheme:

Components	Quiz	Attendance	Mid Term Exam	Presentation/ Assignment/ etc.	End Term Exam
Weightage (%)	10	10	20	10	50

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Better understand the laws of physics governing the motion of physical objects and relationship between force, work and energy.	**PO1, PO2, PO3 & PO6**
CO2	Comprehend the concept of rotational motion of objects, elastic properties of the materials and motion of fluids.	**PO1, PO2, PO3 & PO6**
CO3	Have an understanding of motion under gravitational force of attraction and simple harmonic motion.	**PO1, PO2, PO3 & PO6**

CO4	Gain deeper understanding of Special Theory of Relativity, Lorentz Transformation, Mass energy transformations.	**PO1, PO2, PO3 & PO6**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2	PSO3
ETPH103A	Mechanics	2	2	2			2									3

1=weakly mapped
2= moderately mapped
3=strongly mapped

UCCS155A	Communication Skills	L	T	P	C
Version 1.0		4	0	0	4
Pre-requisites/Exposure	--				
Co-requisites	--				

Course Objectives

1. Understand the basics of Grammar to improve written and oral communication skills.
2. Understand the correct form of English with proficiency
3. Improve student's personality and enhance their self-confidence.
4. Improve professional communication.
5. Enhance academic writing skills.

Course Outcomes

On completion of this course, the students will be able to

CO1. Understand the basics of Grammar to improve written and oral communication skills
CO2. Understand the correct form of English with proficiency
CO3. Improve student's personality and enhance their self-confidence
CO4. Improve professional communication
CO5. Enhance academic writing skills

Catalog Description

This learning program with its practice-based learning tasks will facilitate the learners to enhance their communication skills in a modern and globalized context, enhance their linguistic and communicative competence and hone their interpersonal skills.

Course Content

UNIT I **10 lecture hours**

Introduction to Communication: Importance of Communication Skills, Meaning, Forms & Types of Communication; Process of Communication; Principles of Effective Communication/7Cs, Barriers in Communication (Interpersonal, Intrapersonal and Organizational).

UNIT II **10 lecture hours**

Academic Writing: Précis (Summary – Abstract – Synopsis – Paraphrase – Précis: Methods),

Letter & Résumé (Letter Structure & Elements – Types of letter: Application & Cover - Acknowledgement – Recommendation – Appreciation – Acceptance – Apology – Complaint – Inquiry).Writing a proposal and synopsis. Structure of a research paper. Citations and plagiarism.

UNIT III **10 lecture hours**

Technology-Enabled Communication: Using technology in communication tasks, E-mails, tools for constructing messages, Computer tools for gathering and collecting information; Different virtual medium of communication.

UNIT IV10 lecture hours

Building Vocabulary: Word Formation (by adding suffixes and prefixes); Common Errors; Words Often Confused; One word substitution, Homonyms and Homophones; Antonyms &Synonyms, Phrasal Verbs, Idioms & Proverbs (25 each); Commonly used foreign words(15 in number);

UNIT V **10 lecture hours**

Personality Development: Etiquettes& Manners; Attitude, Self-esteem & Self-reliance; Public Speaking; Work habits (punctuality, prioritizing work, bringing solution to problems), Body Language: Posture, Gesture, Eye Contact, Facial Expressions; Presentation Skills/ Techniques.

Text book [TB]:

1. Kumar, Sanjay and Pushplata.CommunicationSkills.Oxford University Press, 2015.

Reference Books/Materials

1. Mitra, Barun K. Personality Development and Soft Skills. Oxford University Press, 2012.
2. Tickoo, M.L., A. E.Subramanian and P.R.Subramaniam.Intermediate Grammar, Usage and Composition. Orient Blackswan, 1976.
3. Bhaskar, W.W.S., AND Prabhu, NS., " English Through Reading", Publisher: MacMillan,1978
4. Business Correspondence and Report Writing" -Sharma, R.C. and Mohan K. Publisher: Tata McGraw Hill1994
5. Communications in Tourism & Hospitality- Lynn Van Der Wagen, Publisher: HospitalityPress
6. Business Communication-K.K.Sinha
7. Essentials of Business Communication By Marey Ellen Guffey, Publisher: ThompsonPress
8. How to win Friends and Influence People By Dale Carnegie, Publisher: Pocket Books
9. Basic Business Communication By Lesikar&Flatley, Publisher Tata McGraw Hills
10. Body Language By Allan Pease, Publisher SheldonPress

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination
Examination Scheme:

Components	Quiz	Attendance	Mid Term Exam	Presentation/ Assignment/ etc.	End Term Exam
Weightage (%)	10	10	20	10	50

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Understand the basics of Grammar to improve written and oral communication skills	**PO10**
CO2	Understand the correct form of English with proficiency	**PO10**
CO3	Improve student's personality and enhance their self-confidence	**PO12**
CO4	Improve professional communication.	**PO10**
CO5	Enhance academic writing skills	**PO10**

		Engineering Knowledge	Problem analysis	Design/ development of solu	Conduct investigations of com	Modern tool usage	The engineer and society	Environment and sustainabi	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge

				tions	plexproblems			lity								
Course Code	Course Title	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
UCCS155A	Communication Skills										3		3			3

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCS106A	**Clean Coding with Python**	L	T	P	C
Version 1.0	--	3	0	0	3
Pre-requisites/Exposure	--				
Co-requisites	--				

Course Objectives

1. To understand why Python is a useful scripting language for developers.
2. To learn how to design and program Python applications.
3. To learn how to use lists, tuples, and dictionaries in Python programs.
4. To learn how to identify Python object types.
5. To learn how to use indexing and slicing to access data in Python programs.
6. To define the structure and components of a Python program.
7. To learn how to write loops and decision statements in Python.

Course Outcomes

On completion of this course, the students will be able to

CO1.Learn how to write functions and pass arguments in Python.
CO2. Learn how to build and package Python modules for reusability.
CO3. Learn how to read and write files in Python.
CO4. Learn how to design object-oriented programs with Python classes.
CO5. Learn data handling and use cases diagrams.
CO6. Learn how to use class inheritance in Python for reusability.

CO7. Learn how to use exception handling in Python applications for error.

Course Overview: Python is a language with a simple syntax, and a powerful set of libraries. It is an interpreted language, with a rich programming environment, including a robust debugger and profiler. While it is easy for beginners to learn, it is widely used in many scientific areas for data exploration. This course is an introduction to the Python programming language for students without prior programming experience. We cover data types, control flow, object-oriented programming, and graphical user interface-driven applications. The examples and problems used in this course are drawn from diverse areas such as text processing, simple graphics creation and image manipulation, HTML and web programming, and genomics.

Course Content

Unit I: **8 lecture hours**

Introduction to Clean Code: What is Bad Code? Example 1: Avoid, Example 2: for each code, What is Clean Code? , Purpose of Clean Code, Thought of experienced programmers, Intention Revealing Names, Example 1: Poor Variable Names, Example 2: Poor Method Names, Example 3: Variable Name, Make Meaningful Distinctions, Example 1: Usage of Different Words, Example 2: Distinct Names, Use Pronounceable Names, Example 1: Vocal Names, Example 2: Short Form Names, Example 3: Non-Pronounceable Names, Example 4: Compare, Avoid Encodings and Mental Mappings, Difference between smart and professional programmer, Class and Method Names, Function Size Matters, Blocks and Indenting, Do only one thing within a function, One level of abstraction per function, Use Descriptive Name, Example 1: Verbal Names, Function Arguments, Advantages of Having Less Arguments, Command Query Separation, Prefer Exceptions to Returning Error Codes, Extract Try/Catch Blocks, Error Handling Is One Thing, Good Comments, Good Names Can Obviate Comments, Types of Good Comments, Legal Comment, Informative Comment, Explanation of Intent Comment, Clarification Comment, Warning of Consequences Comment, TODO Comments, Amplification Comment, Bad Comments , Mumbling Comments, Redundant Comments, Misleading Comments, Mandated Comments, Journal Comments, Noise Comments, Scary Noise, Commented-Out Code, Too Much

Information, Test Your Knowledge, The Purpose of Formatting, Vertical Formatting, Horizontal Formatting, Team Rules, Exercise 1: Comments and Formatting, Test Your Knowledge, Data Abstraction, Example 1: Concrete Point, Example 2: Abstract Point, Data/Object Antisymmetry, Law of Demeter, Data Transfer Objects.

Unit II: **12 lecture hours**

Introduction to Python: What is Python?, Advantages and disadvantages, Downloading and installing, Which version of Python, Running Python Scripts, Using the interpreter interactively, Using variables, String types: normal, raw and Unicode String operators and expressions, Math operators and expressions, Writing to the screen, Reading from the keyboard, Indenting is significant, The if and elif statements, While Loops, Using List, Dictionaries, Using the for statement, Opening, reading and writing a text file, Using Pandas, the python data analysis library and data frames, Grouping, aggregating and applying, merging and joining, Dealing with syntax errors, Exceptions, Handling exceptions with try/exception.

Unit III: **12 lecture hours**

Data Handling and Use Cases: re Pattern Matching, Parsing Data, Introduction to Regression, Types of Regression, Use Cases, Exploratory data analysis, Correlation Matrix, Visualization using Matplotlib, Implementing linear regression.

Unit IV: **8 lecture hours**

Object Oriented Concepts: Class, Object, Functions, Inheritance. Types of Inheritance, Encapsulation, Polymorphism, Method Overloading and Method Overriding, Data Abstraction, Abstract Classes.

Text Books

1. IBM Material

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination
Examination Scheme:

Components	Quiz	Attendance	Mid Term Exam	Presentation/ Assignment/ etc.	End Term Exam
Weightage (%)	10	10	20	10	50

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs

	Course Outcomes (COs)	Mapped Program Outcomes
CO1	Learn how to write functions and pass arguments in Python.	**PO1**
CO2	Learn how to build and package Python modules for reusability.	**PO4**
CO3	Learn how to read and write files in Python.	**PO5**
CO4	Learn how to design object-oriented programs with Python classes.	**PO2**
CO5	Learn data handling and use cases diagrams.	**PO3**
CO6	Learn how to use class inheritance in Python for reusability.	**PO3**
CO7	Learn how to use exception handling in Python applications for error.	**PO2**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behavior	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2	PSO3

ETCS106A	**Clean Coding with Python**	2	3	3	3	3								3		

1=weakly mapped
2= moderately mapped
3=strongly mapped

UCDM301A	**Disaster Managment**	L	T	P	C
Version 1.0		3	0	0	3
Pre-requisites/Exposure					
Co-requisites	--				

Course Objective:

1. To increase the knowledge and understanding of the disaster phenomenon, its different contextual aspects, impacts and public health consequences.
2. Understanding of the International Strategy for Disaster Reduction (UN-ISDR) and to increase skills and abilities for implementing the Disaster Risk Reduction (DRR) Strategy.
3. To ensure skills and abilities to analyze potential effects of disasters and of the strategies and methods to deliver public health response to avert these effects.
4. To ensure skills and ability to design, implement and evaluate research on disaster.

Course Outcomes:

After completing the program, the student will able to understand

CO1. Capacity to describe, analyze and evaluate the environmental, social, cultural, economic, legal and organizational aspects influencing vulnerabilities and capacities to face disasters.
CO2. The course examines disaster profile of our country and illustrates the role played by various governmental and non- governmental organizations & its effective management.
CO3. It also acquaints learners with the existing legal framework for disaster management.

CO4. Capacity to analyze and evaluate research work on the field of emergencies and disaster while demonstrating insight into the potential and limitations of science, its role in society and people's responsibility for how it is used.

Catalog Description: This course incorporates different types of disasters so that students are well aware of the circumstances around them. We have included one project in the syllabus so that they can thoroughly study the pre & post disastrous situations as well as the role of society in these difficult situations.

Course Content

Unit I: **8 lecture hours**

Introduction to Disasters: Concept and definitions- Disaster, Hazard, vulnerability, resilience, and risks.

Different Types of Disaster: Causes, effects and practical examples for all disasters.

- Natural Disaster: such as Flood, Cyclone, Earthquakes, Landslides etc
- Man-made Disaster: such as Fire, Industrial Pollution, Nuclear Disaster, Epidemic and Biological Disasters, Accidents (Air, Sea, Rail & Road), Structural failures (Building and Bridge), War & Terrorism etc.

Unit II: **8 lecture hours**

Disaster Preparedness and Response Preparedness

- Disaster Preparedness: Concept and Nature
- Disaster Preparedness Plan
- Prediction, Early Warnings and Safety Measures of Disaster.
- Role of Information, Education, Communication, and Training, Role of Government, International and NGO Bodies.
- Role of IT in Disaster Preparedness
- Role of Engineers on Disaster Management.
- Relief and Recovery
- Medical Health Response to Different Disasters

Unit III: **6 lecture hours**

Rehabilitation, Reconstruction and Recovery

- Reconstruction and Rehabilitation as a Means of Development.
- Damage Assessment
- Post Disaster effects and Remedial Measures.

- Creation of Long-term Job Opportunities and Livelihood Options,
- Disaster Resistant House Construction
- Sanitation and Hygiene
- Education and Awareness,
- Dealing with Victims' Psychology,
- Long-term Counter Disaster Planning
- Role of Educational Institute.

Unit IV: **10 lecture hours**

Disaster Management in India

- **Disaster Management Act, 2005:**

  Disaster management framework in India before and after Disaster Management Act, 2005, National Level Nodal Agencies, National Disaster Management Authority

- **Liability for Mass Disaster**

  - Statutory liability
  - Contractual liability
  - Tortiousliability
  - Criminal liability
  - Measure of damages

- **Epidemics Diseases Act, 1897: Main provisions, loopholes.**

- **Project Work**: The project/ field work is meant for students to understand vulnerabilities and to work on reducing disaster risks and to build a culture of safety. Projects must be conceived based on the geographic location and hazard profile of the region where the institute is located.

Reference Books:

- Government of India, Department of Environment, Management of Hazardous Substances Control
- Act and Structure and Functions of Authority Created There under.
- Indian Chemical Manufacturers' Association & Loss Prevention Society of India, Proceedings of the National Seminar on Safety in Road Transportation of Hazardous Materials: (1986).
- Author Title Publication Dr.Mrinalini Pandey Disaster Management Wiley India Pvt. Ltd.

- Tushar Bhattacharya Disaster Science and Management McGraw Hill Education (India) Pvt. Ltd.
- Jagbir Singh Disaster Management: Future Challenges and Opportunities K W Publishers Pvt. Ltd.
- J. P. Singhal Disaster Management Laxmi Publications.
- Shailesh Shukla, ShamnaHussain Biodiversity, Environment and Disaster Management Unique Publications
- C. K. Rajan, NavalePandharinath Earth and Atmospheric Disaster Management: Nature and Manmade B S Publication
- IndianlawInstitute(UpendraBaxiandThomasPaul(ed.),MassDisastersandMultinationalLiability: The Bhopal Case(1986)
- IndianLawInstitute,UpendraBaxi(ed.),EnvironmentProtectionAct:AnAgendaforImplementation (1987)
- Asian Regional Exchange for Prof.Baxi.,Nothing to Lose But our Lives: Empowerment to Oppose
- Industrial Hazards in a Transnational world(1989)
- Guru dip Singh, Environmental Law: International and National Perspectives(1995), Lawman (India)Pvt.Ltd.
- Leela Krishnan, P, The Environmental Law in India, Chapters VIII,IX and X(1999),Butter worths, New Delhi

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination Examination Scheme:

Components	CAT	Mid Term Exam	Attendance/ Class performance	End Term Exam
Weightage (%)	20	20	10	50

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and Pos		
	Course Outcomes (COs)	Mapped Program Outcomes
CO1	Capacity to describe, analyze and evaluate the environmental, social, cultural, economic, legal and organizational aspects influencing vulnerabilities and capacities to face disasters.	PS03

CO2	The course examines disaster profile of our country and illustrates the role played by various governmental and non- governmental organizations & its effective management.	**P03**
CO3	It also acquaints learners with the existing legal framework for disaster management.	**P012**
CO4	Capacity to analyze and evaluate research work on the field of emergencies and disaster while demonstrating insight into the potential and limitations of science, its role in society and people's responsibility for how it is used**.**	**P06**

		Engineering Knowledge	Problem analysis	Design/ development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3

UCDM 301A	Disaster Management			2			3						2			3

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCA164A	**Web Technologies Lab**	L	T	P	C
Version 1.0		0	0	2	1
Pre-requisites/Exposure	Basics of HTML				
Co-requisites	--				

Course Objectives

1. To understand best technologies for solving web client/server problems
2. analyze and design real time web applications
3. use Java script for dynamic effects and to validate form input entry
4. Analyze to Use appropriate client-side or Server-side applications

Course Outcomes

On completion of this course, the students will be able to

CO1. Analyze a web page and identify its elements and attributes. ·

CO2.Create web pages using XHTML and Cascading Style Sheets. ·

CO3.Build dynamic web pages using JavaScript (Client side programming). ·

CO4. Create XML documents and Schemas.

Catalog Description

This course is an introduction to Web site development and the technologies behind it. Students will learn how to design and develop Web pages using current technologies and tools. Topics covered will include the World Wide Web, HTML, Cascading Style Sheets (CSS) and XML.

List of Experiments (Indicative)

1	Write HTML/Java scripts to display your CV in Web Browser	**2 lab hours**
2	Creation and annotation of static web pages using any HTML editor.	**2 lab hours**
3	Write a program to use XML and JavaScript for creation of your homepage.	**2 lab hours**
4	Write a program in XML for creation of DTD which specifies a particular set of rules.	**4 lab hours**
5	Create a Stylesheet in CSS/XSL and display the document in Web Browser	**4 lab hours**
6	Create a Registration Form with Table	**3 lab hours**
7	CSS : Inline Style , Internal Style ,and External Style Sheets	**3 lab hours**
8	JavaScript & HTML: · Use user defined function to get array of values and sort them in ascending order · Demonstrate String and Math Object's predefined methods · Demonstrate Array Objects and Date Object's predefined methods · Exception Handling · Calendar Creation : Display all month · Event Handling · Validation of registration form · Open a Window from the current window ·	**10 lab hours**

	Change color of background at each click of button or refresh of a page · Display calendar for the month and year selected from combo box ·OnMouseover event	
9	XML · Create any catalog · Display the catalog created using CSS or XS	**4 lab hours**

Modes of Evaluation: Quiz/Oral practical oral exam/presentation/projects/Practical Examination

Examination Scheme:

Components	**Quiz**	**Attendance**	**Mid Term Exam**	**Presentation/ Assignment/ etc.**	**End Term Exam**
Weightage (%)	**10**	**10**	**20**	**10**	**50**

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Analyze a web page and identify its elements and attributes. ·	**PO1**
CO2	Create web pages using XHTML and Cascading Style Sheets. ·	**PO4**
CO3	Build dynamic web pages using JavaScript (Client side programming). ·	**PO5**
CO4	Create XML documents and Schemas	**PO2**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2	PSO3
ETCA164A	WEB TECHNOLOGIES LAB	2	2		3	3										3

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCS157A	Clean Coding with Python Lab	L	T	P	C
Version 1.0		0	0	2	1
Pre-requisites/Exposure	--				
Co-requisites	--				

Course Objectives

1. Be fluent in the use of procedural statements — assignments, conditional statements, loops, function calls — and sequences.
2. Be able to design, code, and test small Python programs.
3. Understand the concepts of object-oriented programming as used in Python: classes, subclasses, inheritance, and overriding.
4. Understand the basics of Object Oriented Design.
5. Implement the data structures and use the built-in libraries for efficient codes.

Course Outcomes

On completion of this course, the students will be able to

CO1. Develop solutions to simple computational problems using Python programs.
C02. Solve problems using conditionals and loops in Python. Develop Python programs by defining functions and calling them.
CO3. Use Python lists, tuples and dictionaries for representing compound data.
CO4. Develop Python programs using files.

Catalog Description

Clean coding with python emphasize on principles of software development, style, and testing. Topics include procedures and functions, iteration, recursion, arrays and vectors, strings, an operational model of procedure and function calls, algorithms, exceptions, object-oriented programming, and GUIs (graphical user interfaces). Weekly labs provide guided practice on the computer, with staff present to help. Assignments use graphics and GUIs to help develop fluency and understanding.

Course Content

LIST OF EXPERIMENTS

1	Develop programs to understand the control structures of python	**2 lab hours**
2	Develop programs to implement list	**2 lab hours**

3	Develop programs to implement Dictionary	**2 lab hours**
4	Develop programs to implement tuples	**2 lab hours**
5	Develop programs to implement function with stress on scoping	**2 lab hours**
6	Develop programs to implement classes and objects	**2 lab hours**
7	Develop programs to implement exception handling	**2 lab hours**
8	Develop programs to implement linear search and binary search	**2 lab hours**
9	Develop programs to implement insertion sort	**2 lab hours**
10	Develop programs to implement bubble sort	**2 lab hours**
11	Develop programs to implement quick sort.	**2 lab hours**
12	Develop programs to implement heap sort	**2 lab hours**

Text Books

1. John V Guttag. “Introduction to Computation and Programming Using Python”, Prentice Hall of India.

Reference Books/Materials

1. R. Nageswara Rao, “Core Python Programming”, Dreamtech
2. Wesley J. Chun. “Core Python Programming, Second Edition”, Prentice Hall
3. Michael T. Goodrich, Roberto Tamassia, Michael H. Goldwasser, “Data Structures and Algorithms in Python”, Wiley

4. Kenneth A. Lambert, “Fundamentals of Python,First Programs”, CENGAGE Publication

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination
Examination Scheme:

Components	Quiz	Attendance	Mid Term Exam	Presentation/ Assignment/ etc.	End Term Exam
Weightage (%)	**10**	**10**	**20**	**10**	**50**

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Develop solutions to simple computational problems using Python programs	**PO3**
CO2	Solve problems using conditionals and loops in Python.Develop Python programs by defining functions and calling them	**PO4**
CO3	Use Python lists, tuples and dictionaries for representing compound data	**PO2**
CO4	Develop Python programs using files	**PO3**

		Engineering Knowledge	Problem analysis	Design/development of solutio	Conduct investigations of complex probl	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge

				ns	ems											
Course Code	Course Title	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
ETCS157A	Clean Coding with Python Lab		3	2	3									2		3

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETPH153A	**Mechanics Lab**	L	T	P	C
Version 1.0		0	0	4	2
Pre-requisites/Exposure	-				
Co-requisites	-				

Course Objectives

1. Demonstration cum laboratory sessions on the concepts of mechanics such as moment of inertia, determination of 'g' and elastic constants of materials.
2. Sessions on the review of scientific laboratory report writing, and on experimental data analysis.
3. Expand and exercise the students' physical intuition and thinking process through the experiments.
4. Interpretation of experimental data.

Course Outcomes

On completion of this course, the students will be able to
CO1. Acquire fundamental knowledge of laboratory instruments and their uses.

CO2. Better insight aboutdata collection techniques.
CO3. Better understanding of data interpretation and error analysis..
CO4. Acquire knowledge about the techniques related data analysis and curve fitting.

Catalog Description

This course involves the experimental verification of concepts of mechanics. The course is design to give knowledge how to use basic instruments in laboratory and laboratory experiment protocols. The main focus is on data collection techniques and the data interpretation. For this purpose a series of experiments have been set. The difficulty level of experiments is set easy to moderate level due to introductory physics.

List of Experiments (Indicative)

1	Measurements of length (or diameter) using vernier caliper, screw gauge and travelling microscope.	**2 lab hours**
2	To study the random error in observations.	**2 lab hours**
3	To determine the height of a building using a Sextant.	**2 lab hours**
4	To study the Motion of Spring and calculate (a) Spring constant, (b) g and (c) Modulus of rigidity.	**2 lab hours**
5	To determine the Moment of Inertia of a Flywheel.	**2 lab hours**
6	To determine g and velocity for a freely falling body using Digital Timing Technique	**2 lab hours**
7	To determine Coefficient of Viscosity of water by Capillary Flow Method (Poiseuille's method).	**2 lab hours**
8	To determine the Young's Modulus of a Wire by Optical Lever Method.	**2 lab hours**
9	To determine the Modulus of Rigidity of a Wire by Maxwell's needle.	**2 lab hours**
10	To determine the elastic Constants of a wire by Searle's method.	**2 lab hours**

11	To determine the value of g using Bar Pendulum.	**2 lab hours**
12	To determine the value of g using Kater's Pendulum.	**2 lab hours**

References for Laboratory Work:

- Advanced Practical Physics for students, B. L. Flint and H.T. Worsnop, 1971, Asia Publishing House
- Advanced level Physics Practicals, Michael Nelson and Jon M. Ogborn, 4th Edition, reprinted 1985, Heinemann Educational Publishers
- A Text Book of Practical Physics, I.Prakash & Ramakrishna, 11th Edn, 2011, Kitab Mahal
- Engineering Practical Physics, S.Panigrahi & B.Mallick,2015, Cengage Learning India Pvt.Ltd.
- Practical Physics, G.L. Squires, 2015, 4th Edition, Cambridge University Press.

Modes of Evaluation: Quiz/Oral practical oral exam/presentation/projects/Practical Examination

Examination Scheme:

Components	Quiz	Attendance	Mid Term Exam	Presentation/ Projects/ etc.	End Term Exam
Weightage (%)	**10**	**10**	**20**	**10**	**50**

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Acquire fundamental knowledge of laboratory instruments and their uses.	**PO2**
CO2	Better insight aboutdata collection techniques.	**PO3**
CO3	Better understanding of data interpretation and error analysis.	**PO4**

CO4	Acquire knowledge about the techniques related data analysis and curve fitting.	**PO2 & PO4**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
ETPH153 A	Mechanics Lab		3	2	2											3

1=weakly mapped
2= moderately mapped
3=strongly mapped

Semester II

ETPH102A	Electricity and Magnetism	L	T	P	C
Version 1.0		4	-	-	4
Pre-requisites/Exposure	Basics of Physics				
Co-requisites	--				

Course Objectives

1. The abstraction from forces to fields using the examples of the electric and magnetic fields, with some applications
2. To learn how charges behave through electric circuits.
3. Consolidate the understanding of fundamental concepts in Electricity and Magnetism more rigorously as needed for further studies in physics, engineering and technology.
4. Expand and exercise the students' physical intuition and thinking process through the understanding of the theory and application of this knowledge to the solution of practical problems

Course Outcomes

On completion of this course, the students will be able to

CO1. Acquire fundamental knowledge of electrostatic interaction using Gauss Law and able to apply on physical systems.
CO2. Better insight about magnetic and dielectric behaviour of materials.
CO3. Better understanding of electrical circuits/theorems which enhances problem solving approach.
CO4. Develop the ability to correlates the daily life phenomenon to physics using mathematical tools.

Catalog Description

This course imparts the basic concepts of Physics. The course is design to point to a plausible physical origin of simple electromagnetic phenomena in nature, based on what the candidate has learned in the course about fundamental laws and concepts in electricity and magnetism. The course of Electricity and Magnetism help organizing the data in variety of ways to solve the problem efficiently. The course is focused on theoretical discussions of Electricity and Magnetism and applications of discussed phenomenon. It also discusses about daily life physics like magnetism, current etc.

Course Content

Unit I: **10 Lecture hours**

Electric field: Electric field lines. Electric flux. Gauss' Law with applications to charge distributions with spherical, cylindrical and planar symmetry.

Conservative nature of Electrostatic Field .Electrostatic Potential. Laplace's and Poisson equations. The Uniqueness Theorem. Potential and Electric Field of a dipole. Force and Torque on a dipole.

Unit II: **10 Lecture hours**

Electrostatic energy of system of charges.Electrostatic energy of a charged sphere.Conductors in an electrostatic Field. Surface charge and force on a conductor. Capacitance of a system of charged conductors.Parallel-plate capacitor.Capacitance of an isolated conductor. Method of Images and its application to: (1) Plane Infinite Sheet and (2) Sphere.

Dielectric Properties of Matter: Electric Field in matter. Polarization, Polarization Charges.Electrical Susceptibility and Dielectric Constant. Capacitor (parallel plate, spherical, cylindrical) filled with dielectric. Displacement vector D. Relations between E, P and D. Gauss' Law in dielectrics.

Unit III: **10 Lecture hours**

Magnetic Field: Magnetic force between current elements and definition of Magnetic Field B. Biot-Savart's Law and its simple applications: straight wire and circular loop. Current Loop as a Magnetic Dipole and its Dipole Moment (Analogy with Electric Dipole). Ampere's Circuital Law and its application to (1) Solenoid and (2) Toroid. Properties of B: curl and divergence. Vector Potential. Magnetic Force on (1) point charge (2) current carrying wire (3) between current elements. Torque on a current loop in a uniform Magnetic Field.

Magnetic Properties of Matter: Magnetization vector (M). Magnetic Intensity (H). Magnetic Susceptibility and permeability. Relation between B, H, M. Ferromagnetism. B-H curve and hysteresis.

Unit IV: **10 Lecture hours**

Electromagnetic Induction: Faraday's Law. Lenz's Law. Self Inductance and Mutual Inductance. Reciprocity Theorem. Energy stored in a Magnetic Field. Introduction to Maxwell's Equations. Charge Conservation and Displacement current.

Electrical Circuits: AC Circuits: Kirchhoff's laws for AC circuits. Complex Reactance and Impedance. Series LCR Circuit: (1) Resonance, (2) Power Dissipation and (3) Quality Factor, and (4) Band Width. Parallel LCR Circuit.

Network theorems: Ideal Constant-voltage and Constant-current Sources. Network Theorems: The venin theorem, Norton theorem, Superposition theorem, Reciprocity theorem, Maximum Power Transfer theorem. Applications to dc circuits.

Ballistic Galvanometer: Torque on a current Loop. Ballistic Galvanometer: Current and Charge Sensitivity. Electromagnetic damping. Logarithmic damping.CDR.

Text Books

1. Physics for Scientists and Engineers (6th Ed.), Raymond A. Serway and John W. Jewett, Thomson Brooks (2004).
2. Engineering Physics Theory and Practical, A. K. Katiyar and C. K. Pandey, Wiley (2015)

Reference Books/Materials

1. Introduction to Electrodynamics (3rd Indian reprint), D.J. Griffiths,., Pearson Education (2003).
2. Electricity, Magnetism & Electromagnetic Theory, S. Mahajan and Choudhury, 2012, Tata McGraw

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination
Examination Scheme:

Components	Quiz	Attendance	Mid Term Exam	Presentation/ Assignment/ etc.	End Term Exam
Weightage (%)	10	10	20	10	50

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	Mapped Program Outcomes
CO1	Acquire fundamental knowledge of electrostatic interaction using Gauss Law and able to apply on physical systems.	PO1& PO2

CO2	Better insight about magnetic and dielectric behaviour of materials.	**PO4**
CO3	Better understanding of electrical circuits/theorems which enhancesproblem solving approach.	**PO6**
CO4	Develop the ability to correlates the daily life phenomenon to physics using mathematical tools.	**PO7 & PO8**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Application of Concepts	Ethics	Analysis
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
ETPH102 A	Electricity & Magnetism	2	2		2		2	2	3					3		2

1=weakly mapped
2= moderately mapped
3=strongly mapped

UCES125A	Environmental Studies	L	T	P	C
Version 1.0		3	0	0	3
Pre-requisites/Exposure	Basics of Environment				
Co-requisites	--				

Course Objectives

1. To aware the students about the environment.
2. To learn the students concepts and methods from ecological and physical sciences and their application in environmental problem solving.
3. To think across and beyond existing disciplinary boundaries, mindful of the diverse forms of knowledge and experience that arises from human interactions with the world around them.
4. Communicate clearly and competently matters of environmental concern and understanding to a variety of audiences in appropriate forms.

Course Outcomes

On completion of this course, the students will be able to

CO1. To comprehend and become responsive regarding environmental issues.

CO2.Acquire the techniques to protect our mother earth, as without a clean, healthy, aesthetically beautiful, safe and secure environment no specie can survive and sustain.

CO3. Enable the students to discuss their concern at national and international level with respect to formulate protection acts and sustainable developments policies.

CO4.To know that the rapid industrialization, crazy consumerism and over-exploitation of natural resources have resulted in degradation of earth at all levels.

CO5. Become consciousness about healthy and safe environment.

Catalog Description

This course imparts the basic concepts of environment which enable them to solve basic problems related to their surroundings. This course helps them to get an idea adverse effect of industrialization, population and degradation of natural resources on the environment. The course introduces the concepts of renewable and non-renewable resources.

Course Content

UNIT I **10 Lectures**

Environment and Natural Resources:

Multidisciplinary nature of environmental sciences; Scope and importance; Need for public awareness.

Land resources; land use change; Land degradation, soil erosion and desertification.

Deforestation: Causes and impacts due to mining, dam building on environment, forests, biodiversity and tribal populations.

Water: Use and over-exploitation of surface and ground water, floods, droughts, conflicts over water (international & inter-state).

Energy resources: Renewable and non- renewable energy sources, use of alternate energy sources, growing energy needs, case studies.

UNIT II **10 Lectures**

Ecosystems and Biodiversity:

Ecosystem: Definition and Structure and function of ecosystem; Energy flow in an ecosystem: food chains, food webs and ecological succession.

Case studies of the following ecosystems:

a) Forest ecosystem

b) Grassland ecosystem

c) Desert ecosystem

d) Aquatic ecosystems (ponds, streams, lakes, rivers, oceans, estuaries)

Biological diversity: genetic, species and ecosystem diversity; Biogeographic zones of India; Biodiversity patterns and global biodiversity hot spots ; India as a mega-biodiversity nation; Endangered and endemic species of India; Threats to biodiversity: Habitat loss, poaching of wildlife, man-wildlife conflicts, biological invasions; Conservation of biodiversity: In-situ and Ex-situ conservation of biodiversity; Ecosystem and biodiversity services: Ecological, economic, social, ethical, aesthetic and Informational value.

UNIT III **10 Lectures**

Environmental Pollution and Environmental Policies:

Environmental pollution: types, causes, effects and controls; Air, water, soil and noise pollution Nuclear hazards and human health risks; Solid waste management: Control measures of urban and industrial waste; Pollution case studies.

Sustainability and sustainable development; Climate change, global warming, ozone layer depletion, acid rain and impacts on human communities and agriculture; Environment Laws: Environment Protection Act; Air (Prevention & Control of Pollution) Act; Water (Prevention and control of Pollution) Act; Wildlife Protection Act; Forest Conservation Act; Nature reserves, tribal populations and rights, and human wildlife conflicts in Indian context.

UNIT IV **10 Lectures**

Human Communities and the Environment and Field work:
Human population growth: Impacts on environment, human health and welfare; Resettlement and rehabilitation of project affected persons; case studies; Disaster management: floods, earthquake, cyclones and landslides; Environmental movements: Chipko, Silent valley, Bishnois of Rajasthan; Environmental ethics: Role of Indian and other religions and cultures in environmental conservation; Environmental communication and public awareness, case studies (e.g., CNG vehicles in Delhi).
Visit to an area to document environmental assets: river/ forest/ flora/fauna, etc.
Visit to a local polluted site-Urban/Rural/Industrial/Agricultural. Study of common plants, insects, birds and basic principles of identification. Study of simple ecosystems-pond, river, Delhi Ridge, etc.

Text Books

1. Kaushik and Kaushik, Environmental Studies, New Age International Publishers (P) Ltd. New Delhi.

Reference Books/Materials

1. A.K. De, Environmental Chemistry, New Age International Publishers (P) Ltd. New Delhi.
2. S.E. Manahan, Environmental Chemistry, CRC Press.
3. S.S Dara and D.D. Mishra, Environmental Chemistry and Pollution Control, S.Chand& Company Ltd, New Delhi.
4. R. Gadi, S. Rattan, S. Mohapatra, Environmental Studies Kataria Publishers, New Delhi.

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination
Examination Scheme:

Components	Quiz	Attendance	Mid Term Exam	Presentation/ Assignment/ etc.	End Term Exam
Weightage (%)	10	10	20	10	50

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	Mapped Program Outcomes
CO1	The learners will be able to comprehend and become responsive regarding environmental	PO7

	issues.	
CO2	Students will acquire the techniques to protect our mother earth, as without a clean, healthy, aesthetically beautiful, safe and secure environment no specie can survive and sustain.	**PO8**
CO3	It enables the students to discuss their concern at national and international level with respect to formulate protection acts and sustainable developments policies.	**PO10**
CO4	Students come to know that the rapid industrialization, crazy consumerism and over-exploitation of natural resources have resulted in degradation of earth at all levels.	**PO6**
CO5	Students become consciousness about healthy and safe environment.	**PO7**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2	PSO3

UCES125 A	Environmental Studies						2	3	3		3				1	2

1=weakly mapped
2= moderately mapped
3=strongly mapped.

ETCS307A	**Database Management Systems**	L	T	P	C
Version 1.0		3	1	0	4
Pre-requisites/Exposure	Basics of Data Base				
Co-requisites	--				

Course Objectives

1. To understand the different issues involved in the design and implementation of a database system.

2. To study the physical and logical database designs, database modeling, relational, hierarchical, and network models.

3. To understand and use data manipulation language to query, update, and manage a database.

4. To develop an understanding of essential DBMS concepts such as: database security, integrity, concurrency, distributed database, and intelligent database, Client/Server (Database Server), Data Warehousing.

5. To design and build a simple database system and demonstrate competence with the fundamental tasks involved with modeling, designing, and implementing a DBMS.

6. For a given query write relational algebra expressions for that query and optimize the developed expression.

Course Outcomes

On completion of this course, the students will be able to

CO1. Independently understand basic database technology.
CO2.Describe the fundamental elements of relational database management systems
CO3. Explain the basic concepts of relational data model, entity-relationship model, relational database design, relational algebra and SQL.
CO4.Design ER-models to represent simple database application scenarios
CO5. Convert the ER-model to relational tables, populate relational database and formulate SQL queries on data.
CO6.Improve the database design by normalization.
CO7. Familiar with basic database storage structures and access techniques: file and page organizations, indexing methods including B tree, and hashing.

CO8. Students will be able to work in a group on the design, and implementation of a database system project.

Catalog Description

Database Management Systems (DBMS) are vital components of modern information systems. Database applications are pervasive and range in size from small in-memory databases to terra bytes or even larger in various applications domains. The course focuses on the fundamentals of knowledge base and relational database management systems, and the current developments in database theory and their practice. The course reviews topics such as conceptual data modelling, relational data model, relational query languages, relational database design and transaction processing and current technologies.

Course Content

Unit I: **12 lecture hours**

Database system architecture: Data Abstraction, Data Independence, Data Definition Language (DDL), Data Manipulation Language (DML). Data models: Entity-relationship model, network model, relational and object oriented data models, integrity constraints, data manipulation operations.

Unit II: **8 lecture hours**

Relational query languages: Relational algebra, Tuple and domain relational calculus, SQL3, DDL and DML constructs, Open source and Commercial DBMS - MYSQL, ORACLE, DB2, SQL server. Relational database design: Domain and data dependency, Armstrong's axioms, Normal forms, Dependency preservation, Lossless design. Query processing and optimization: Evaluation

of relational algebra expressions, Query equivalence, Join strategies, Query optimization algorithms.

Unit III: 12 lecture hours

Storage strategies: Indices, B-trees, hashing, Transaction processing: Concurrency control, ACID property, Serializability of scheduling, Locking and timestamp based schedulers, Multi-version and optimistic Concurrency Control schemes, Database recovery

Unit IV: 8 lecture hours

Database Security: Authentication, Authorization and access control, DAC, MAC and RBAC models, Intrusion detection, SQL injection. Advanced topics: Object oriented and object relational databases, Logical databases, Web databases, Distributed databases, Data warehousing and data mining.

Text Books

1."Database System Concepts", 6th Edition by Abraham Silberschatz, Henry F. Korth, S. Sudarshan, McGraw-Hill.

2. "Principles of Database and Knowledge – Base Systems", Vol 1 by J.D. Ullman, Computer Science Press.

Reference Books/Materials

1. "Fundamentals of Database Systems", R. Elmasri and S. Navathe, Pearson Education

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination
Examination Scheme:

Components	Quiz	Attendance	Mid Term Exam	Presentation/ Assignment/ etc.	End Term Exam
Weightage (%)	10	10	20	10	50

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	Mapped Program Outcomes
CO1	Independently understand basic database technology.	PO2

CO2	Describe the fundamental elements of relational database management systems	**PO3**
CO3	Explain the basic concepts of relational data model, entity-relationship model, relational database design, relational algebra and SQL.	**PO4**
CO4	Design ER-models to represent simple database application scenarios	**PO5**
CO5	Convert the ER-model to relational tables, populate relational database and formulate SQL queries on data.	**PO4**
CO6	Improve the database design by normalization.	**PO4**
CO7	Familiar with basic database storage structures and access techniques: file and page organizations, indexing methods including B tree, and hashing.	**PO9**
CO8	Students will be able to work in a group on the design, and implementation of a database system project.	**PSO3**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3

ETCS307 A	Database Management Systems		2	3	3	3				3						3

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCS112A	**Object Oriented Programming**	L	T	P	C
Version 1.0		3	1	0	4
Pre-requisites/Exposure	Basics of Computer Programming				
Co-requisites	--				

Course Objectives

This course introduces the concepts of object-oriented programming to students with a background in the procedural paradigm. The course begins with a brief review of control structures and data types with emphasis on structured data types and array processing. It then moves on to introduce the object-oriented programming paradigm, focusing on the definition and use of classes along with the fundamentals of object-oriented design. Other topics include an overview of programming language principles, simple analysis of algorithms, basic searching and sorting techniques, event-driven programming, memory management and an introduction to software engineering issues.

Course Outcomes

On completion of this course, the students will be able to

CO1 Explain the steps in creating an executable program for a computer, including the intermediate representations and their purpose.
CO2. Manipulate binary patterns and understand the use of binary to represent numbers. CO 3. Apply good programming style and understand the impact of style on developing and maintaining programs. CO4. Effectively use a version control system and the Linux command line tools for

incremental development.
CO5. Explain the benefits of object-oriented design and understand when it is an appropriate methodology to use.
CO6. Design object-oriented solutions for small systems involving multiple objects.
CO7. Identify the relative merits of different algorithmic designs.

Catalog Description

This is a course in which you learn computer programming concepts that are fundamental in nearly any computer programming language. These concepts can then be used in other courses to help you create computer applications that can be used to solve real-world problems

Course Content

Unit I: **12 lecture hours**

Introduction: Introducing Object-Oriented Approach related to other paradigms (functional, data decomposition), Characteristics of Object-Oriented Languages.
Basic terms and ideas: Abstraction, Encapsulation, Information hiding, Inheritance, Polymorphism, Review of C, Difference between C and C++, Cin, Cout, new, delete operators.

Unit II: **8 lecture hours**

Classes and Objects: Abstract data types, Object & classes, attributes, methods, C++ class declaration, State identity and behavior of an object, Constructors and destructors, instantiation of objects, Default parameter value, Copy Constructor, Static Class Data, Constant Classes, C++ garbage collection, dynamic memory allocation.

Unit III: **12 lecture hours**

Inheritance and Polymorphism: Inheritance, Types of Inheritance, Class hierarchy, derivation – public, private & protected, Agrégations, composition vs classification hiérarchies, Polymorphism, Type of Polymorphism – Compile time and runtime, Method polymorphism, Polymorphism by parameter, Operator overloading, Parametric polymorphism, Generic function – template function, function name overloading, Overriding inheritance methods.

Unit IV: **8 lecture hours**

Files and Exception Handling: Persistent objects, Streams and files, Namespaces, Exception handling, Generic Classes Standard Template Library: Standard Template Library, Overview of

Standard Template Library, Containers, Algorithms, Iterates, Other STL Elements, The Container Classes, General Theory of Operation, Vectors.

Text Books

A.R. Venugopal, Rajjkumar, T. Ravishanker "Mastering C++", TMH

2. R. Lafore, "Object Oriented Programming using C++", BPB Publications

3. Schildt Herbert, "C++ Programming", 2nd Edition, Wiley DreamTech

Reference Books/Materials

1. D. Parasons, "Object Oriented Programming with C++", BPB Publication
2. Steven C. Lawlor, "The Art of Programming Computer Science with C++", Vikas Publication
3. YashwantKanethkar, "Object Oriented Programming using C++", BPB

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination
Examination Scheme:

Components	Quiz	Attendance	Mid Term Exam	Presentation/ Assignment/ etc.	End Term Exam
Weightage (%)	10	10	20	10	50

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	CO1 Explain the steps in creating an executable program for a computer, including the intermediate representations and their purpose.	**PO2**
CO2	CO2. Manipulate binary patterns and understand the use of binary to represent numbers. CO 3. Apply good programming style and understand the impact of style on developing and maintaining programs. CO4. Effectively use a version control system and the Linux command line tools for incremental development.	**PO3**
CO3	CO5. Explain the benefits of object-oriented design and understand when it is an appropriate methodology to use.	**PO4**
CO4	CO6. Design object-oriented solutions for small systems	**PO5**

	involving multiple objects.	
CO5	CO7. Identify the relative merits of different algorithmic designs.	**PO4**
CO6	CO1 Explain the steps in creating an executable program for a computer, including the intermediate representations and their purpose.	**PO4**
CO7	CO2. Manipulate binary patterns and understand the use of binary to represent numbers. CO 3. Apply good programming style and understand the impact of style on developing and maintaining programs. CO4. Effectively use a version control system and the Linux command line tools for incremental development.	**PO9, PSO 3**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2
ETCS112A	Object oriented programming		2	3	3	3				3					

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCS355A	Database Managemet Systems Lab	L	T	P	C
Version 1.0		-	-	2	1
Pre-requisites/Exposure	--				
Co-requisites	--				

Course Objectives

1. To explain basic database concepts, applications, data models, schemas and instances.
2. To demonstrate the use of constraints and relational algebra operations.
3. To facilitate students in Database design.
4. To familiarize issues of concurrency control and transaction management.

Course Outcomes

On completion of this course, the students will be able to:-
CO1. Apply the basic concepts of Database Systems and Applications.
CO2. Use the basics of SQL and construct queries using SQL in database creation and interaction.
CO3. Design a commercial relational database system (Oracle, MySQL) by writing SQL using the system.
CO4. Analyze and Select storage and recovery techniques of database system.

Catalog Description

This course introduces the core principles and techniques required in the design and implementation of database systems. This introductory application-oriented course covers the relational database systems RDBMS - the predominant system for business scientific and engineering applications at present. It includes Entity-Relational model, Normalization, Relational model, Relational algebra, and data access queries as well as an introduction to SQL. It also covers essential DBMS concepts such as: Transaction Processing, Concurrency Control and Recovery. It also provides students with theoretical knowledge and practical skills in the use of databases and database management systems in information technology applications.

Course Content

List of Experiments

S.No	Experiment	No of Hours
1	Design a Database and create required tables. For e.g. Bank, College Database	4

2	Apply the constraints like Primary Key, Foreign key, NOT NULL to the tables.	2
3	Write a SQL statement for implementing ALTER, UPDATE and DELETE.	2
4	Write the queries to implement the joins.	4
5	Write the queries for implementing the following functions: MAX (), MIN (), AVG (), COUNT ().	2
6	Write the queries to implement the concept of Integrity constrains	4
7	Write the queries to create the views.	2
8	Perform the queries for triggers.	4
9	Perform the following operation for demonstrating the insertion, updating and deletion using the referential integrity constraints.	2
10	Do some more practice based on your class work.	2

Text Books

1. "Database System Concepts", 6th Edition by Abraham Silberschatz, Henry F. Korth, S. Sudarshan, McGraw-Hill.

Reference Books/Materials

1. "Principles of Database and Knowledge – Base Systems", Vol 1 by J.D. Ullman, Computer Science Press.
2. "Fundamentals of Database Systems", R. Elmasri and S. Navathe, Pearson Education.

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination
Examination Scheme:

Components	Quiz	Attendance	Mid Term Exam	Presentation/ Assignment/ etc.	End Term Exam
Weightage (%)	10	10	20	10	50

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	Mapped Program Outcomes

CO1	Apply the basic concepts of Database Systems and Applications	**PO5**
CO2	Use the basics of SQL and construct queries using SQL in database creation and interaction	**PO3**
CO3	Design a commercial relational database system (Oracle, MySQL) by writing SQL using the system	**PO3**
CO4	Analyze and Select storage and recovery techniques of database system.	**PO2**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
ETCS355A	Database Management Systems Lab		3	3		2								3		

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCS166A	Object Oriented Programming Lab	L	T	P	C
Version 1.0		0	0	2	1
Pre-requisites/Exposure	Practical learning				
Co-requisites	--				

Course Objectives

This course will give the learner an insight into how everything can be considered an object and how simply we can write code to implement it. It helps us in making programming relatable to real world, as everything around us can be an object (having properties and functionality) Object-oriented programming aims to implement real world entities like inheritance, hiding, polymorphism etc in programming. The main aim of OOP is to bind together the data and the functions that operates on them so that no other part of code can access this data except that function.

Course Outcomes

On completion of this course, the students will be able to

CO1 Understand fundamentals of programming such as variables, conditional and iterative execution, methods, etc.
CO2 Understand fundamentals of object-oriented programming including defining classes, invoking methods, using class libraries, etc.
CO3 Be aware of the important topics and principles of software development.
CO4 Develop the ability to write a computer program to solve specified problems.

Catalog Description

This course emphasizes solving problems using the language, and introduces standard programming techniques like alternation, iteration and recursion. It will briefly glimpse the basics of software engineering practices like modularization, commenting, and naming conventions which help in collaborating and programming in teams. This course is enabled the students to formulate algorithms for arithmetic and logical problems, convert these algorithms to C language programs. It also aims on using arrays, pointers and structures to formulate algorithms and programs. In addition to that, apply programming to solve matrix addition and multiplication problems and searching and sorting problems.

Course Content

1	Raising a number n to a power p is the same as multiplying n by itself p times. Write a function called power ( ) that takes a double	**2 lab hours**

	value for n and an int value for p, and returns theresult as double value. Use a default argument of 2 for p, so that if this argument is omitted, the number will be squared. Write a main ( ) function that gets values from the user to test this function.	
2	A point on the two dimensional plane can be represented by two numbers: an X coordinate and a Y coordinate. For example, (4,5) represents a point 4 units to the right of the origin along the X axis and 5 units up the Y axis. The sum of two points can be defined as a new point whose X coordinate is the sum of the X coordinates of the points and whose Y coordinate is the sum of their Y coordinates. Write a program that uses a structure called point to model a point. Define three points, and have the user input values to two of them. Than set the third point equal to the sum of the other two, and display the value of the new point. Interaction with the program might look like this: Enter coordinates for P1: 3 4 Enter coordinates for P2: 5 7 Coordinates of P1 + P2 are : 8, 11	**2 lab hours**
3	Create the equivalent of a four function calculator. The program should request the user to enter a number, an operator, and another number. It should then carry out the specified arithmetical operation: adding, subtracting, multiplying, or dividing the two numbers. (It should use a switch statement to select the operation). Finally it should display the result. When it finishes the calculation, the program should ask if the user wants to do another calculation. The response can be 'Y' or 'N'. Some sample interaction with the program might look like this. Enter first number, operator, second number: 10/ 3 Answer = 3.333333 Do another (Y/ N)? Y Enter first number, operator, second number 12 + 100 Answer = 112	**2 lab hours**
4	A phone number, such as (212) 767-8900, can be thought of as having three parts: the area code (212), the exchange (767) and the number (8900). Write a program that uses a structure to store these three parts of a phone number separately. Call the structure phone. Create two structure variables of type phone. Initialize one, and have the user input a number for the other one. Then display both numbers. The interchange might look like this: Enter your area code, exchange, and number: 415 555 1212 My number is (212) 767-8900 Your number is (415) 555-1212	**2 lab hours**
5	Create two classes DM and DB which store the value of distances. DM stores distances in meters and centimeters and DB in feet and inches. Write a program that can read values for the class objects	**2 lab hours**

	and add one object of DM with another object of DB. Use a friend function to carry out the addition operation. The object that stores the results maybe a DM object or DB object, depending on the units in which the results are required. The display should be in the format of feet and inches or meters and centimeters depending on the object on display.	
6	Create a class rational which represents a numerical value by two double valuesNUMERATOR & DENOMINATOR. Include the following public member Functions: • constructor with no arguments (default). • constructor with two arguments. • void reduce( ) that reduces the rational number by eliminating the highest common factor between the numerator and denominator. • Overload + operator to add two rational number. • Overload >> operator to enable input through cin.• Overload << operator to enable output through cout. Write a main ( ) to test all the functions in the class.	**4 lab hours**
7	Consider the following class definition class father { protected : int age; public; father (int x) {age = x;} virtual void iam ( ) { cout<< "I AM THE FATHER, my age is : "<< age<< end1:} }; Derive the two classes son and daughter from the above class and for each, define iam ( ) to write our similar but appropriate messages. You should also define suitable constructors for these classes. Now, write a main ( ) that creates objects of the three classes and then calls iam ( ) for them. Declare pointer to father. Successively, assign addresses of objects of the two derived classes to this pointer and in each case, call iam ( ) through the pointer to demonstrate polymorphism in action.	**4 lab hours**
8	Write a program that creates a binary file by reading the data for the students from the terminal. The data of each student consist of roll no., name (a string of 30 or lesser no. of characters) and marks.	**4 lab hours**
9	A hospital wants to create a database regarding its indoor patients. The information to store include a) Name of the patient b) Date of admissionDisease d) Date of discharge Create a structure to store the date (year, month and date as its members). Create a base class to store the above information. The member function should include functions to enter information and display a list of all the patients in the database. Create a derived class to store the age of the patients. List the information about all the to store the age of the patients. List the information about all the pediatric patients	**4 lab hours**

	(less than twelve years in age).	
10	Make a class Employee with a name and salary. Make a class Manager inherit from Employee. Add an instance variable, named department, of type string. Supply a method to toString that prints the manager's name, department and salary. Make a class Executive inherit from Manager. Supply a method to String that prints the string "Executive" followed by the information stored in the Manager superclass object. Supply a test program that tests these classes and methods.	**2 lab hours**
11	Imagine a tollbooth with a class called toll Booth. The two data items are a type unsigned int to hold the total number of cars, and a type double to hold the total amount of money collected. A constructor initializes both these to 0. A member function called payingCar ( ) increments the car total and adds 0.50 to the cash total. Another function, called nopayCar ( ), increments the car total but adds nothing to the cash total. Finally, a member function called displays the two totals. Include a program to test this class. This program should allow the user to push one key to count a paying car, and another to count a nonpaying car. Pushing the ESC kay should cause the program to print out the total cars and total cash and then exit.	**2 lab hours**
12	Write a function called reversit ( ) that reverses a string (an array of char). Use for loop that swaps the first and last characters, then the second and next to last characters and so on. The string should be passed to reversit ( ) as an argument. Write a program to exercise reversit ( ). The program should get a string from the user, call reversit ( ), and print out the result. Use an input method that allows embedded blanks. Test the program with Napoleon's famous phrase, "Able was I ere I saw Elba)".	**2 lab hours**
13	. Create some objects of the string class, and put them in a Deque-some at the head of the Deque and some at the tail. Display the contents of the Deque using the forEach ( ) function and a user written display function. Then search the Deque for a particular string, using the first That ( ) function and display any strings that match. Finally remove all the items from the Deque using the getLeft ( ) function and display each item. Notice the order in which the items are displayed: Using getLeft ( ), those inserted on the left (head) of the Deque are removed in "last in first out" order while those put on the right side are removed in "first in first out" order. The opposite would be true if getRight ( ) were used.	**2 lab hours**

14	Create a base class called shape. Use this class to store two double type values that could be used to compute the area of figures. Derive two specific classes called triangle and rectangle from the base shape. Add to the base class, a member function get_data ( ) to initialize base class data Members and another member function display_area ( ) to compute and display the area of figures. Make display_area ( ) as a virtual function and redefine this function in the derived classes to suit their requirements. Using these three classes, design a program that will accept dimensions of a triangle or a rectangle interactively and display the area. Remember the two values given as input will be treated as lengths of two sides in the case of rectangles and as base and height in the case of triangles and used as follows: Area of rectangle = x * y Area of triangle = ½ * x * y	**2 lab hours**

Modes of Evaluation: Quiz/Oral practical oral exam/presentation/projects/Practical Examination

Examination Scheme:

Components	**Quiz**	**Attendance**	**Mid Term Exam**	**Presentation/ Assignment/ etc.**	**End Term Exam**
Weightage (%)	**10**	**10**	**20**	**10**	**50**

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Understand fundamentals of programming such as variables, conditional and iterative execution, methods, etc.	**PO2**
CO2	Understand fundamentals of object-oriented programming including defining classes, invoking methods, using class libraries, etc.	**PO3**
CO3	Be aware of the important topics and principles of software development.	**PO5, PSO3,**

CO4	Develop the ability to write a computer program to solve specified problems.	**PO9**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
ETCS166A	Object oriented programming Lab		2	3		3				3						3

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETPH152A	Electricity and Magnetism Lab	L	T	P	C
Version 1.0		0	0	4	2
Pre-requisites/Exposure	-				
Co-requisites	-				

Course Objectives

1. Dedicated demonstration cum laboratory sessions on the construction, functioning and uses of different electrical bridge circuits, and electrical devices like the ballistic galvanometer. To learn how charges behave through electric circuits.
2. Sessions on the review of scientific laboratory report writing, and on experimental data analysis.
3. Expand and exercise the students' physical intuition and thinking process through the experiments.
4. Interpretation of experimental data

Course Outcomes

On completion of this course, the students will be able to:
CO1. Acquire fundamental knowledge of laboratory instruments and their uses.
CO2. Better insight aboutdata collection techniques.
CO3. Better understanding of data interpretation and error analysis.
CO4. Acquire knowledge about the techniques related data analysis and curve fitting.

Catalog Description

This course imparts the basic concepts of experimental physics. The course is design to give knowledge how to use basic instruments in laboratory and laboratory experiment protocols. The main focus is on data collection techniques and the data interpretation. For this purpose a series of experiments have been set. The difficulty level of experiments is set easy to moderate level due to introductory physics.

List of Experiments (Indicative)

1	Use a Multimeter for measuring (a) Resistances, (b) AC and DC Voltages, (c) DC Current, (d) Capacitances, and (e) Checking electrical fuses.	**2 lab hours**

2	To study the characteristics of a series RC Circuit.	**2 lab hours**
3	To determine an unknown Low Resistance using Potentiometer.	
4	To determine an unknown Low Resistance using Carey Foster's Bridge.	**2 lab hours**
5	To compare capacitances using De'Sauty's bridge.	**2 lab hours**
6	Measurement of field strength B and its variation in a solenoid (determine dB/dx)	**2 lab hours**
7	To verify the Thevenin and Norton theorems.	**2 lab hours**
8	To verify the Superposition, and Maximum power transfer theorems.	**2 lab hours**
9	To determine self inductance of a coil by Anderson's bridge.	**2 lab hours**
10	To study response curve of a Series LCR circuit and determine its (a) Resonant frequency, (b) Impedance at resonance, (c) Quality factor Q, and (d) Band width.	**2 lab hours**
11	To study the response curve of a parallel LCR circuit and determine its (a) Anti-resonantfrequency and (b) Quality factor Q.	**2 lab hours**
12	Measurement of charge and current sensitivity and CDR of Ballistic Galvanometer	**2 lab hours**
13	Determine a high resistance by leakage method using Ballistic Galvanometer.	**2 lab hours**
14	To determine self-inductance of a coil by Rayleigh's method.	**2 lab hours**
15	To determine the mutual inductance of two coils by Absolute method.	**2 lab hours**

Text Books

- Physics for Scientists and Engineers (6th Ed.), Raymond A. Serway and John W. Jewett, Thomson Brooks (2004).
- Engineering Physics Theory and Practical, A. K. Katiyar and C. K. Pandey, Wiley (2015)

Reference Books/Materials

- Introduction to Electrodynamics (3rd Indian reprint), D.J. Griffiths,., Pearson Education (2003).
- Electricity, Magnetism & Electromagnetic Theory, S. Mahajan and Choudhury, 2012, Tata McGraw

Modes of Evaluation: Quiz/Oral practical oral exam/presentation/projects/Practical Examination

Examination Scheme:

Components	Quiz	Attendance	Mid Term Exam	Presentation/ Projects/ etc.	End Term Exam
Weightage (%)	**10**	**10**	**20**	**10**	**50**

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Acquire fundamental knowledge of laboratory instruments and their uses.	**PO2**
CO2	Better insight aboutdata collection techniques.	**PO3**
CO3	Better understanding of data interpretation and error analysis.	**PO4**
CO4	Acquire knowledge about the techniques related data analysis and curve fitting.	**PO2 & PO4**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2	PSO3
ETPH153A	Mechanics Lab		3	2	2											3

1=weakly mapped
2= moderately mapped
3=strongly mapped

Semester III

ETCS 321A	Java Programming	L	T	P	C
Version 1.0		3	1	0	4
Pre-requisites/Exposure	NIL				
Co-requisites	--				

Course Objectives

1. Explain the concepts of object oriented paradigms to solve problems.
2. Appraise the concept of reusable software components using inheritance, packages and interfaces
3. Create scalable applications that can robustly handle errors and exceptions in runtime applications
4. Designing applications using pre-built frameworks.

Course Outcomes

On completion of this course, the students will be able to

CO1. Learn syntax of Java Programming Language and implement applications using it.
CO2. Recognize features of object-oriented design such as encapsulation, polymorphism inheritance and composition of systems based on object identity.
CO3. Articulate re-usable programming components using Abstract Class, Interfaces and other permitted ways in packages.
CO4. Apply access control mechanism to safeguard the data and functions that can be applied by the object.
CO5. Understand multithreading and evaluate exception handing to create new applications.
CO6. Design GUI applications using pre-built frameworks available in Java.

Catalog Description

Java's unique architecture enables programmers to develop applications that can run across multiple platforms seamlessly and reliably. In this hands-on course, students gain extensive experience with Java and its object-oriented features. Students learn to create robust console and GUI applications and store and retrieve data from relational databases.

Course Content

Unit I: **12 lecture hours**

Introduction to Java: Introduction to Java: Importance and features of Java, Keywords, constants, variables and Data Types, Operators and Expressions, Decision Making, Branching and Looping: if..else, switch,?: operator, while, do, for statements, labeled loops, jump statements: break, continue return. Introducing classes, objects and methods: defining a class, adding variables and methods, creating objects, constructors, class inheritance.

Unit II: 9 lecture hours

Arrays and Strings: Creating an array, one and two dimensional arrays, string array and methods, Classes: String and String Buffer classes, Wrapper classes: Basics types, using super, Multilevel hierarchy, abstract and final classes, Object class, Packages and interfaces, Access protection, Extending Interfaces, packages.

Unit III: 9 lecture hours

Exceptional Handling: Fundamentals exception types, uncaught exceptions, throw, throw, final, built in exception, creating your own exceptions, Multithreaded Programming: Fundamentals, Java thread model: priorities, synchronization, messaging, thread classes, Run able interface, inter thread Communication, suspending, resuming and stopping threads.

Unit IV: 10 lecture hours

Input/output Programming: Basics Streams, Byte and Character Stream, predefined streams, Reading and writing from console and files. Using Standard Java Packages (lang, util, io, net).

Event Handling: Different Mechanism, the Delegation Event Model, Event Classes, Event Listner Interfaces, Adapter and Inner Classes.

Text Books

1. Cay S. Horstmann, "Core Java Volume – I Fundamentals", Pearson.

Reference Books/Materials

1. Herbert Schildt, "Java – The Complete Reference", Oracle Press.

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination
Examination Scheme:

Components	Quiz	Attendance	Mid Term Exam	Presentation/ Assignment/ etc.	End Term Exam
Weightage (%)	10	10	20	10	50

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Learn to the syntax of Java Programming Language and implement applications in it.	**PO2**
CO2	Recognize features of object-oriented design such as encapsulation, polymorphism, inheritance and composition of systems based on object identity.	**PO3**
CO3	Articulate re-usable programming components using Abstract Class, Interfaces and other permitted ways in packages.	**PO5**
CO4	Apply access control mechanism to safeguard the data and functions that can be applied by the object	**PO8**
CO5	Understand multithreading and evaluate exception handing to create new applications.	**PO1**
CO6	Design GUI applications using pre-built frameworks available in Java.	**PO9**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS	PS	PS

														O1	O2	O3
ETCS321A	Java Programming	2	3	3		2			2	3				3		2

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCS211A	**Operating Systems**	L	T	P	C
Version 1.0		3	1	0	4
Pre-requisites/Exposure	Computer Organization & Architecture				
Co-requisites	--				

Course Objectives

1. To learn the mechanisms of OS to handle processes and threads and their communication.
2. To learn the mechanisms involved in memory management in contemporary OS
3. To gain knowledge on distributed operating system concepts that includes architecture, Mutual exclusion algorithms, deadlock detection algorithms and agreement protocols
4. To know the components and management aspects of concurrency management
5. To learn to implement simple OS mechanisms

Course Outcomes

On completion of this course, the students will be able to:

CO1. Describe the important computer system resources and the role of operating system in their management policies and algorithms.

CO2. To understand various functions, structures and history of operating systems and should be able to specify objectives of modern operating systems and describe how operating systems have evolved over time.
CO3. Understanding of design issues associated with operating systems.
CO4. Understand various process management concepts including scheduling, synchronization, and deadlocks.
CO5. To understand concepts of memory management including virtual memory.
CO6. To understand issues related to file system interface and implementation, disk management

Catalog Description

This course will provide an introduction to the internal operation of modern operating systems. In particular, the course will cover processes and threads, mutual exclusion, CPU scheduling, deadlock, memory management, and file systems.

Course Content

Unit I: **6 lecture hours**

Introduction: Concept of Operating Systems, Generations of Operating systems, Types of Operating Systems, OS Services, System Calls, Structure of an OS-Layered, Monolithic, Microkernel Operating Systems, Concept of Virtual Machine. Case study on UNIX and WINDOWS Operating System.

Unit II: **12 lecture hours**

Processes: Definition, Process Relationship, Different states of a Process, Process State transitions, Process Control Block (PCB), Context switching

Thread: Definition, Various states, Benefits of threads, Types of threads, Concept of multithreads,

Process Scheduling: Foundation and Scheduling objectives, Types of Schedulers, Scheduling criteria: CPU utilization, Throughput, Turnaround Time, Waiting Time, Response Time;

Scheduling algorithms: Pre-emptive and Non-preemptive, FCFS, SJF, RR; Multiprocessor

scheduling: Real Time scheduling: RM and EDF.

Unit III: **12 lecture hours**

Memory Management: Basic concept, Logical and Physical address map, Memory allocation: Contiguous Memory allocation – Fixed and variable partition–Internal and External fragmentation and Compaction; Paging: Principle of operation – Page allocation – Hardware

support for paging, Protection and sharing, Disadvantages of paging.

Virtual Memory: Basics of Virtual Memory – Hardware and control structures – Locality of reference, Page fault, Working Set, Dirty page/Dirty bit – Demand paging, Page Replacement algorithms: Optimal, First in First Out (FIFO), Second Chance (SC), Not recently used (NRU) and Least Recently used (LRU).

File Management: Concept of File, Access methods, File types, File operation, Directory structure, File System structure, Allocation methods (contiguous, linked, indexed), Free- space management (bit vector, linked list, grouping), directory implementation (linear list, hash table), efficiency and performance.

Unit IV: **12 lecture hours**

Process-Synchronization & Deadlocks: Inter-process Communication: Critical Section, Race Conditions, Mutual Exclusion, Hardware Solution, Strict Alternation, Peterson's Solution, The Producer\ Consumer Problem, Semaphores, Event Counters, Monitors, Message Passing, Classical IPC Problems: Reader's & Writer Problem, Dinning Philosopher Problem etc. Definition of Deadlocks, Necessary and sufficient conditions for Deadlock, Deadlock Prevention, Deadlock Avoidance: Banker's algorithm, Deadlock detection and Recovery.

I/O Systems: I/O devices, Device controllers, Direct memory access Principles of I/O Software: Goals of Interrupt handlers, Device drivers, Device independent I/O software, Secondary-Storage Structure: Disk structure, Disk scheduling algorithms

Text Books

1. Silbersachatz and Galvin, "Operating System Concepts", Pearson

Reference Books/Materials

1. Tannenbaum, "Operating Systems", PHI, 4th Edition.
2. William Stallings, "Operating Systems Internals and Design Principles", PHI
3. HallMadnick, J. Donovan, "Operating Systems", Tata McGraw Hill.
4. W. Tomasi, "Electronic Communication Systems" Pearson Education, 5th Edition

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination
Examination Scheme:

Components	Quiz	Attendance	Mid Term Exam	Presentation/ Assignment/ etc.	End Term Exam
Weightage (%)	10	10	20	10	50

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Describe the important computer system resources and the role of operating system in their management policies and algorithms.	**PO1**
CO2	To understand various functions, structures and history of operating systems and should be able to specify objectives of modern operating systems and describe how operating systems have evolved over time.	**PO1**
CO3	Understanding of design issues associated with operating systems.	**PO3**
CO4	Understand various process management concepts including scheduling, synchronization, and deadlocks.	**PO4**
CO5	To understand concepts of memory management including virtual memory.	**PO5**
CO6	To understand issues related to file system interface and implementation, disk management.	**PO3**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PS	PSO3

															O 2	
ETCS 211A	Operating Systems	2		3	3	3								2		

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETEC 210A	**Digital Electronics**	L	T	P	C
Version 1.0		3	1	0	4
Pre-requisites/Exposure					
Co-requisites	--				

Course Objectives

1. To acquire the basic knowledge of digital logic levels and application of knowledge to understand digital electronics circuits.
2. To understand number representation and conversion between different representation in digital electronic circuits.
3. To analyze logic processes and implement logical operations using combinational logic circuits.
4. To understand characteristics of memory and their classification.
5. To understand concepts of sequential circuits and to analyze sequential systems in terms of state machines.

Course Outcomes

On completion of this course, the students will be able to

CO1. Create the appropriate truth table from a description of a combinational logic function.
CO2. Create a gate-level implementation of a combinational logic function described by a truth table using and/or/not gates, multiplexers or ROMs, and analyse its timing behaviour.
CO3. Create a state transition diagram from a description of a sequential logic function and then convert the diagram into an implementation of a finite-state machine with the appropriate combinational and sequential components.
CO4. Describe the operation and timing constraints for latches and registers.
CO5. Draw a circuit diagram for a sequential logic circuit and analyse its timing properties (input setup and hold times, minimum clock period, output propagation delays).
CO6. Evaluate combinational and sequential logic designs using various metrics: switching speed, throughput/latency, gate count and area, energy dissipation and power.

Catalogue Description

This course helps the student to develop a digital logic and apply it to solve real life problems and will able to analyze, design and implement combinational logic circuits and sequential logic circuits.

Course Contents

Unit I: **10 lecture hours**

UNIT – I

Number Systems and Codes: Review of number systems, BCD codes and arithmetic, Gray code, self-complimenting codes, Error detection and correction principles.
Digital Circuits: Switching algebra & simplification of Boolean expressions. De Morgan‘s Theorem. Implementations of Boolean expressions using logic gates

Unit II: **10 lecture hours**

Combinational Logic Design: Combinational circuit analysis and synthesis, Techniques for minimization of Boolean functions such as Karnaugh map, VEM and Quine-Mc Cluskey methods. Design of arithmetic circuits, code convertors, multiplexers, demultiplexers, encoders, decoders & comparators. Parity generators and checker.
Introduction to Sequential Logic: Need for sequential circuits, Binary cell, Latches and flip-flops. RS, JK, Master-Slave JK, D & T flip flops.

Unit III: **10 lecture hours**

Synchronous Sequential Circuit Design: Fundamentals of Synchronous sequential circuits, Classification of synchronous machines, Analysis of Synchronous Sequential circuits, Design of

Synchronous and Asynchronous Counters, Shift registers & Ring counters, Analysis and design of Finite State Machines. Timing issues in synchronous circuits.
Logic Families**:** Performance metrics of logic gates, Basic Transistor-Transistor Logic and CMOS logic.

Unit IV: 10 lecture hours

Asynchronous Sequential Circuits: Fundamentals of Asynchronous Sequential circuits. Analysis and design of Asynchronous Sequential circuits. Pulse mode and Fundamental-mode Circuits. Cycles, Races and Hazards in asynchronous circuits.

Text Books

1. William I. Fletcher, ―An Engineering approach to Digital Design‖, Prentice Hall of India
2. C.H.Roth, ―Fundamentals of Logic Design, Thomson
3. Morris Mano, "Digital Design", PHI, 2nd Ed.

Reference Books/Materials

1. J. Nagrath, "Electronics, Analog & Digital", PHI.
2. B. S. Nai, "Digital Electronics and Logic Design", PHI.
3. Balabanian and Carlson, "Digital Logic Design Principles", Wiley Pub.

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination
Examination Scheme:

Components	Quiz	Attendance	Mid Term Exam	Presentation/ Assignment/ etc.	End Term Exam
Weightage (%)	10	10	20	10	50

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Create the appropriate truth table from a description of a combinational logic function.	**PO1**
CO2	Create a gate-level implementation of a combinational logic function described by a truth table using and/or/not gates, multiplexers or ROMs, and analyze its timing behavior.	**PO2**

CO3	Create a state transition diagram from a description of a sequential logic function and then convert the diagram into an implementation of a finite-state machine with the appropriate combinational and sequential components.	**PO3**
CO4	Describe the operation and timing constraints for latches and registers.	**PO4**
CO5	Draw a circuit diagram for a sequential logic circuit and analyze its timing properties (input setup and hold times, minimum clock period, output propagation delays).	**PO5**
CO6	Evaluate combinational and sequential logic designs using various metrics: switching speed, throughput/latency, gate count and area, energy dissipation and power.	**PO3**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2	PSO3
ETEC210A	Digital Electronics	2	2	3	3	3								2		3

1=weakly mapped
2= moderately mapped
3= strongly mapped

ETCS219A	Foundation Of Computer Systems	L	T	P	C
Version 1.0		3	1	0	4
Pre-requisites/Exposure	Some concepts from basic math – algebra, geometry, pre-calculus				
Co-requisites	--				

Course Objectives

1. Use mathematically correct terminology and notation.
2. Construct correct direct and indirect proofs.
3. Use division into cases in a proof.
4. Use counterexamples.
5. Apply logical reasoning to solve a variety of problems.

Course Outcomes

On completion of this course, the students will be able to:

CO1. Acquire an understanding set theory, functions, and relations.

CO2. Develop the given problem as graph networks and solve with techniques of graph theory.

CO3. Understanding the language of mathematical logic and expressing statements in terms of logic.

CO4. Derive the solution for a given problem using deductive logic and prove the solution based on logical inference.

CO5. Gaining insight into applications of discrete mathematics to various practical problems.

Catalog Description

The course is an introduction to discrete mathematics as a foundation to work within the fields of computer science, information technologies, and software development.

Course Content

Unit I: **10 lecture hours**

Set Theory: Introduction to set theory, Set operations, Algebra of sets, Duality, Finite and Infinite sets, Classes of sets, Power Sets, Multi sets, Cartesian Product, Representation of relations, Types of relation, Equivalence relations and partitions , Partial ordering relations and lattices Function and its types, Composition of function and relations, Cardinality and inverse relations

Unit II: **12 lecture hours**

Graphs And Trees: Introduction to graphs, Directed and Undirected graphs, Homomorphic and Isomorphic graphs, Subgraphs, Cut points and Bridges, Multigraph and Weighted graph, Paths and circuits, Shortest path in weighted graphs, Eurelian path and circuits, Hamilton paths and circuits, Planar graphs, Euler's formula, Trees, Spanning trees, Binary trees and its traversals.

Unit III: **12 lecture hours**

Propositional logic: Basic operations: AND(^), OR(v), NOT(~), Truth value of a compound statement, propositions, tautologies, contradictions, Validity of Arguments

Group theory: Definition and examples of a monoid, Semigroup, Groups and rings, Homomorphism, Isomorphism and Auto morphism, Subgroups and Normal subgroups, Cyclic groups, Co-Sets, Lagrange's theorem.

Unit IV: **10 lecture hours**

Recursion and Recurrence Relation: linear recurrence relation with constant coefficients, Homogeneous solutions, Solutions, Total solution of a recurrence relation using generating functions.

Techniques Of Counting: Permutations with and without repetition, Combination.

Text Books

1. Keneth H. Rosen, "Discrete Mathematics and Its Applications", TMH.
2. C.L. Liu, "Elements of Discrete Mathematics", TMH.

Reference Books/Materials

1. Kolman, Busby & Ross, "Discrete Mathematical Structures", PHI.
2. NarsinghDeo, "Graph Theory with Application to Engineering and Computer Science", PHI.
3. J. P. Trembly& P. Manohar, "Discrete Mathematical Structures with Applications to

Computer Science", McGraw Hill.

4. Vinay Kumar, "Discrete Mathematics", BPB Publications.

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination
Examination Scheme:

Components	Quiz	Attendance	Mid Term Exam	Presentation/ Assignment/ etc.	End Term Exam
Weightage (%)	10	10	20	10	50

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Acquire an understanding set theory, functions, and relations.	**PO1**
CO2	Develop the given problem as graph networks and solve with techniques of graph theory.	**PO2**
CO3	Understanding the language of mathematical logic and expressing statements in terms of logic.	**PO1**
CO4	Derive the solution for a given problem using deductive logic and prove the solution based on logical inference.	**PO3**
CO5	Gaining insight into applications of discrete mathematics to various practical problems.	**PO3**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2	PSO3
ETCS219A	Foundation of Computer Systems	3	3	2										2		1

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCS217A	Data Structures	L	T	P	C
Version 1.0		3	1	0	4
Pre-requisites/Exposure	Basics of Computer Programming				
Co-requisites	--				

Course Objectives

1. To be able to compute the efficiency of algorithms in terms of time and space complexities.
2. To understand concepts of searching and sorting algorithms.
3. Using various data structures viz. stacks, queues, linked list, trees and graphs to develop efficient algorithms through efficient representation of data and operations that can be applied.
4. To enable them to develop algorithms for solving problem by applying concepts of data structures.

Course Outcomes

On completion of this course, the students will be able to

CO1. Analyze the algorithms to determine the time and computation complexity and justify the correctness.
C02. Implement a given Search problem (Linear Search and Binary Search).
CO3. Write algorithms concerning various data structures like Stack, Queue, Linked list, Graph search and traversal techniques and analyze the same to determine the time and computation complexity.
CO4. Write an algorithm for Selection Sort, Bubble Sort, Insertion Sort, Quick Sort, Merge Sort, Heap sort and compare their performance in term of Space and time complexity.

Catalog Description

This course imparts the basic concepts of data structures and algorithms. It enables them to write algorithms for solving problems with the help of fundamental data structures. The course of data structures help organizing the data in variety of ways to solve the problem efficiently. The course introduces the basic concepts about stacks, queues, lists, trees and graphs. It also discusses about daily problems like searching and sorting techniques

Course Content

Unit I: **8 lecture hours**

Introduction to Data Structures: Definition of data structures and abstract data types, Static and Dynamic implementations, Examples and real life applications; Arrays**:** ordered lists, representation of arrays, sparse matrices, polynomial arithmetic

Running time: Analysis of Algorithms and their complexities: Time Complexities, Big – Oh - notation, Running Times, Best Case, Worst Case, Average Case, Factors depends on running time, Introduction to Recursion, Divide and Conquer Algorithm, Time & Space Tradeoff.

Unit II: **12 lecture hours**

The Stacks: ADT Stack and its operation, Array based implementation of stacks, Linked List based implementation of stacks, Examples: Infix, postfix, prefix representation, Conversions, Applications, Algorithms and their complexities

Queues and Lists: ADT Queue and its operation, Array based implementation of linear Queues, Circular implementation of Queues, Linked Lists: Singly linked lists: Representation of linked lists in memory, Traversing, Searching, Insertion into, Deletion from linked list Linked List implementation of Queues and Stacks Lists, Straight / circular implementation of doubly linked Queues / Lists, Priority Queues, Applications, Algorithms and their complexities

Unit III: **12 lecture hours**

Trees: Basic Terminology, Binary Trees and their representation, expression evaluation, Complete Binary trees, Extended binary trees, traversing binary trees, Searching, Insertion and Deletion in binary search trees (with and without recursion), AVL trees, Threaded trees, B+ trees, algorithms and their analysis.

Graphs: Terminology and Representations, Graphs & Multigraphs, Directed Graphs, Sequential representation of graphs, Adjacency matrices, Transversal Connected Component and Spanning trees, Shortest path, algorithms and their analysis.

Unit IV: **8 lecture hours**

Sorting Algorithms: Introduction, Sorting by exchange, selection sort, insertion sort, Bubble sort, Straight selection sort, Efficiency of above algorithms, Shell sort, Performance of shell sort, Merge sort, Merging of sorted arrays& Algorithms; Quick sort Algorithm analysis, heap sort: Heap Construction, Heap sort, bottom – up, Top – down Heap sort approach;

Searching Algorithms: Straight Sequential Search, Binary Search (recursive & non–recursive Algorithm

Text Books

1. E. Horowitz and S. Sahani, “Fundamentals of Data Structures”, Galgotia Book source Pvt. Ltd.

2. R. L. Kruse, B. P. Leung, C. L. Tondo, “Data Structures and program design in C”, PHI

Reference Books/Materials

1. Schaum’s outline series, “Data Structure”, McGraw Hills.
2. Y. Langsamet. al., “Data Structures using C and C++”, PHI.

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination
Examination Scheme:

Components	Quiz	Attendance	Mid Term Exam	Presentation/ Assignment/ etc.	End Term Exam
Weightage (%)	10	10	20	10	50

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	Mapped Program Outcomes
CO1	Analyze the algorithms to determine the time and computation complexity	PO1
CO2	Implement a given Search problem (Linear Search and Binary Search).	PO4
CO3	Write algorithms concerning various data structures	PO5
CO4	Write an algorithm for internal and external sorting	PO2

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2	PSO3
ETCS217A	Data Structures	2	2		3	3								3		3

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCA365A	**Linux Environment Lab**	L	T	P	C
Version 1.0		0	0	2	1
Pre-requisites/Exposure	-				
Co-requisites	-				

Course Objectives

The objective of this course is to impart necessary and practical knowledge concerning basic Linux usage.

1. To implement some standard Linux utilities such as ls.cpetc
2. To write shell script programs to solve problems.
3. To learn basics of system administration

Course Outcomes

On completion of this course, the students will be able to

CO1. Understanding the basic set of commands and utilities in Linux/UNIX systems
CO2. Able to create file handling utilities by using Linux shell environment
CO3. Evaluate the concept of shell scripting programs
CO4. Obtain a foundation for System Administration

Catalog Description

This course will provide you with a basic introduction to Linuxskills The student will learn how a Linux system is organized, and will demonstrate introductory system administration tasks. The student will be able to reasons why Linux and the open source development model are so important in today's computing environment.

List of Experiments (Indicative)

1	Installing Linux Operating System	**2 lab hours**
2	**Exploring the System:** Starting Up and changing run levels, Using the man utility, Using built-in help switches for commands, Using Auto completion	**2 lab hours**
3	Using cd**,** Using pwd**,** Using mkdir**,** Using rmdir,	
4	Using Touch, Using ls**,**Using mv**,** Using cp**,** Using cat, Using Redirection**,** rm**,** Using vi	**2 lab hours**
5	Searching for files: grep, frep and similar commands	**2 lab hours**
6	Preamble, Virtual terminals**,** Setting up a basic display**,**X clients**,** Window Managers**,** Display Manager**,** xinit and startx**,** system-config-display	**2 lab hours**
7	Manually creating a new user, Manually creating a new groups, automatically creating a new user, automatically creating new groups, using sticky bits, share the file between users and groups..	**2 lab hours**
8	Installing, Querying and Uninstalling Packages, Third party tools, Building Software from Source	**2 lab hours**

9	Determining device type, Creating devices, mounting and umounting devices	**2 lab hours**
10	Different kind of shells (c shell, bash shell, korn shell and others), A simple Script, Using variables in scripts	**2 lab hours**
11	Shell Script: Using Control Structures	**4 lab hours**

Modes of Evaluation: Quiz/Oral practical oral exam/presentation/projects/Practical Examination

Examination Scheme:

Components	**Quiz**	**Attendance**	**Mid Term Exam**	**Presentation/ Assignment/ etc.**	**End Term Exam**
Weightage (%)	**10**	**10**	**20**	**10**	**50**

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Understanding the basic set of commands and utilities in Linux/UNIX systems	**PO5**
CO2	Able to create file handling utilities by using Linux shell environment.	**PO6**
CO3	Evaluate the concept of shell scripting programs	**PO3**
CO4	Obtain a foundation for System Administration	**PO12**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behavior	Knowledge
Course Code	Course Title	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
ETCA365A	Linux Environment Lab			2		3	2						3			2

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCS257A	Data Structures Lab	L	T	P	C
Version 1.0		0	0	2	1
Pre-requisites/Exposure	Basics of Computer Programming				
Co-requisites	--				

Course Objectives

1. To be able to compute the efficiency of algorithms in terms of time and space complexities.
2. To understand concepts of searching and sorting algorithms.
3. Using various data structures viz. stacks, queues, linked list, trees and graphs to develop efficient algorithms through efficient representation of data and operations that can be applied.
4. To enable them to develop algorithms for solving problem by applying concepts of data structures.

Course Outcomes

On completion of this course, the students will be able to

CO1. Analyze the algorithms to determine the time and computation complexity and justify the correctness.
CO2. Implement a given Search problem (Linear Search and Binary Search).
CO3. Write algorithms concerning various data structures like Stack, Queue, Linked list, Graph search and traversal techniques and analyze the same to determine the time and computation complexity.
CO4. Write an algorithm for Selection Sort, Bubble Sort, Insertion Sort, Quick Sort, Merge Sort, Heap sort and compare their performance in term of Space and time complexity.

Catalog Description

This course complements ETCS 217A. It enables them to write algorithms for solving problems with the help of fundamental data structures. The list of experiments help organizing the data in variety of ways using data structures and to solve the given problem efficiently. It also discusses about daily problems like searching and sorting techniques

List of Experiments (Indicative)

1	Write a program for multiplication and transpose of array.	**2 lab hours**
2	Write a program to compute the transpose of a sparse matrix	**2 lab hours**
3	Write a program to implement push and pop operation in Stack.	**2 lab hours**
4	Write a program to convert a Infix notation to post fix notation using stacks	**2 lab hours**
5	Write a program to evaluate postfix notation using stacks	**2 lab hours**
6	Write a program to implement a linear queue	**2 lab hours**
7	Write a program for swapping two numbers using call by value and call by reference strategies.	**2 lab hours**
8	Write a program to insert and delete a node in linked list. The number of nodes to inserted and deleted should be governed by user.	**3 lab hours**
9	Write a program to implement a linear search arrays and linked list.	**2 lab hours**
10	Using iteration and recursion concepts write programs for finding the element in the array using the Binary search method.	**2 lab hours**
11	Write the programs to implement bubble sort.	**2 lab hours**

12	Write a program using iteration and recursion concepts for quick sort.	**2 lab hours**
13	Write a program to implement merge sort.	**2 lab hours**
14	Write a program to simulate various tree traversal techniques.	**3 lab hours**
15	Write a program to simulate various BFS and DFS.	**4 lab hours**

Modes of Evaluation: Quiz/Oral practical oral exam/presentation/projects/Practical Examination

Examination Scheme:

Components	**Quiz**	**Attendance**	**Mid Term Exam**	**Presentation/ Assignment/ etc.**	**End Term Exam**
Weightage (%)	**10**	**10**	**20**	**10**	**50**

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Analyze the algorithms to determine the time and computation complexity	**PO1**
CO2	Implement a given Search problem (Linear Search and Binary Search).	**PO4**
CO3	Write algorithms concerning various data structures	**PO5**
CO4	Write an algorithm for internal and external sorting	**PO2**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2	PSO3
ETCS257A	Data Structures Lab	2	2		3	3								3		3

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCS 255A	**Operating Systems Lab**	L	T	P	C
Version 1.0		0	0	2	1
Pre-requisites/Exposure	Computer Organization & Architecture				
Co-requisites	--				

Course Objectives

1. To learn the mechanisms of OS to handle processes and threads and their communication.
2. To learn the mechanisms involved in memory management in contemporary OS

3. To gain knowledge on distributed operating system concepts that includes architecture, Mutual exclusion algorithms, deadlock detection algorithms and agreement protocols
4. To know the components and management aspects of concurrency management
5. To learn to implement simple OS mechanisms

Course Outcomes

On completion of this course, the students will be able to:

CO1. Describe the important computer system resources and the role of operating system in their management policies and algorithms.
CO2. To understand various functions, structures and history of operating systems and should be able to specify objectives of modern operating systems and describe how operating systems have evolved over time.
CO3. Understanding of design issues associated with operating systems.
CO4. Understand various process management concepts including scheduling, synchronization, and deadlocks.
CO5. To understand concepts of memory management including virtual memory.
CO6. To understand issues related to file system interface and implementation, disk management

Catalog Description

Based on theory subject **ETCS 211A**, the following experiments are to be performed. It enables them to write algorithms for solving problems with the help of fundamental operating systems.

List of Experiments (Indicative)

1	Write a C program to simulate the following non-preemptive CPU scheduling algorithms to find turnaround time and waiting time. a) FCFS b) SJF c) Round Robin (pre-emptive) d) Priority	**4 lab hours**
2	Write a C program to simulate multi-level queue scheduling algorithm considering the following scenario. All the processes in the system are divided into two categories – system processes and user processes. System processes are to be given higher priority than user processes. Use FCFS scheduling for the processes in each queue.	**2 lab hours**
3	Given the list of processes, their CPU burst times and arrival times, display/print the Gantt chart for Priority and Round robin. For each of the scheduling policies, compute and print the average waiting time and average turnaround time.	**4 lab hours**

4	Write a C program to simulate the following file allocation strategies. a) Sequential b) Indexed c) Linked	**4 lab hours**
5	Write a C program to simulate the MVT and MFT memory management techniques.	**4 lab hours**
6	Write a C program to simulate the following contiguous memory allocation techniques a) Worst-fit b) Best-fit c) First-fit	**2 lab hours**
7	Write a C program to simulate paging technique of memory management	**4 lab hours**
8	Write a C program to simulate the following file organization techniques a) Single level directory b) Two level directory c) Hierarchical	**4 lab hours**
9	Write a C program to simulate Bankers algorithm for the purpose of deadlock avoidance.	**4 lab hours**
10	Write a C program to simulate page replacement algorithms a) FIFO b) LRU c) LFU	**2 lab hours**

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination

Examination Scheme:

Components	**Quiz**	**Attendance**	**Mid Term Exam**	**Presentation/ Assignment/ etc.**	**End Term Exam**
Weightage (%)	**10**	**10**	**20**	**10**	**50**

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and Pos		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Describe the important computer system resources and the role of operating system in their management policies and algorithms.	**PO1**
CO2	To understand various functions, structures and history of operating systems and should be able to specify objectives of modern operating systems and describe how operating systems have evolved over time.	**PO1**
CO3	Understanding of design issues associated with operating systems.	**PO3**

CO4	Understand various process management concepts including scheduling, synchronization, and deadlocks.	**PO4**
CO5	To understand concepts of memory management including virtual memory.	**PO5**
CO6	To understand issues related to file system interface and implementation, disk management.	**PO3**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
ETCS 255A	Operating Systems Lab	2		3	3	3								2		

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCS361A	Java Programming Lab	L	T	P	C
Version 1.0		0	0	2	1
Pre-requisites/Exposure	Basics of Computer Programming				
Co-requisites	--				

Course Objectives

1. Explain the concepts of object oriented paradigms to solve problems.
2. Appraise the concept of reusable software components using inheritance, packages and interfaces
3. Create scalable applications that can robustly handle errors and exceptions in runtime applications
4. Designing applications using pre-built frameworks.

Course Outcomes

On completion of this course, the students will be able to

CO1. Learn to the syntax of Java Programming Language and implement applications in it.
CO2. Recognize features of object-oriented design such as encapsulation, polymorphism inheritance and composition of systems based on object identity.
CO3. Articulate re-usable programming components using Abstract Class, Interfaces and other permitted ways in packages.
CO4. Apply access control mechanism to safeguard the data and functions that can be applied by the object.
CO5. Understand multithreading and evaluate exception handing to create new applications.
CO6. Design GUI applications using pre-built frameworks available in Java.

Catalog Description

This course complements ETCS 323A. It enables them to write algorithms for solving problems with the help of fundamental data structures. The list of experiments help organizing the data in variety of ways using data structures and to solve the given problem efficiently. It also discusses about daily problems like searching and sorting techniques

List of Experiments (Indicative)

1	Create a java program to implement stack and queue.	**2 lab hours**
2	Write a java program to demonstrate dynamic polymorphism.	**2 lab hours**
3	Write a java program to implement various shapes using Abstract	**2 lab hours**

	class	
4	Write a java program to demonstrate interfaces.	**2 lab hours**
5	Write a java program to show multithreaded producer and consumer application.	**2 lab hours**
6	Create a java programs that make use of all the 5 exception keywords.	**4 lab hours**
7	Convert the content of a given file into the uppercase content of the same file.	**4 lab hours**
8	Develop a scientific calculator using swings.	**4 lab hours**
9	Create a servlet that uses Cookies to store the number of times a user has visited your servlet.	**4 lab hours**
10	Create a simple java bean having bound and constrained properties.	**4 lab hours**

Modes of Evaluation: Quiz/Oral practical oral exam/presentation/projects/Practical Examination

Examination Scheme:

Components	**Quiz**	**Attendance**	**Mid Term Exam**	**Presentation/ Assignment/ etc.**	**End Term Exam**
Weightage (%)	**10**	**10**	**20**	**10**	**50**

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Learn to the syntax of Java Programming Language and implement applications in it.	**PO2**
CO2	Recognize features of object-oriented design such as encapsulation, polymorphism, inheritance and composition of systems based on object identity.	**PO3**
CO3	Articulate re-usable programming components using Abstract Class, Interfaces and other permitted ways in packages.	**PO5**
CO4	Apply access control mechanism to safeguard the data and functions that can be applied by the object	**PO8**

CO5	Understand multithreading and evaluate exception handing to create new applications.	**PO1**
CO6	Design GUI applications using pre-built frameworks available in Java.	**PO9**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behavior	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2	PSO3
ETCS361A	Java Programming Lab	2	3	3		2			2	3				3		2

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETEC256A	Digital Electronics Lab	L	T	P	C
Version 1.0		0	0	2	1
Pre-requisites/Exposure					
Co-requisites	--				

Course Objectives:

1. Explain the elements of digital system abstractions such as digital representations of information, digital logic, Boolean algebra, state elements and finite state machine (FSMs).
2. Design simple digital systems based on these digital abstractions, using the "digital paradigm" including discrete sampled information.
3. Use the "tools of the trade": basic instruments, devices and design tools.
4. Work in a design team that can propose, design, successfully implement and report on a digital systems project.
5. Communicate the purpose and results of a design project in written and oral presentations.

Course Outcomes:

On completion of this course, the students will be able to

CO1. Identify relevant information to supplement to the Digital Electronic ETEC210A course.
CO2. Construct basic combinational circuits and verify their functionalities
CO3. To understand the basic digital circuits and to verify their operation.
CO4. To understand the concepts of flip flops, registers and counters.
CO5. To understand how gates are the basic building blocks for digital world.

Catalogue Description:
Labs on digital logic, PALs, flip-flops, timing, counters, synchronization, and finite-state machines prepare students for the design and implementation of a final project of their choice, e.g., games, music, digital filters, wireless communications, graphics, etc. Extensive use of Verilog for describing and implementing digital logic designs. Students engage in extensive written and oral communication exercises

Course Content

List of experiments:

- Introduction to digital electronics lab- nomenclature of digital ICs, specifications, study of the data sheet, concept of Vcc and ground, verification of the truth tables of logic gates using TTL ICs.

- Implementation of the given Boolean function using logic gates in both SOP and POS forms.
- Verification of state tables of RS, JK, T and D flip-flops using NAND & NOR gates.
- Implementation and verification of Decoder/De-multiplexer and Encoder using logic gates.
- Implementation of 4x1 multiplexer using logic gates.
- Implementation of 4-bit parallel adder using 7483 IC.
- Design, and verify the 4-bit synchronous counter.
- Design, and verify the 4-bit asynchronous counter.
- Static and Dynamic Characteristic of NAND and Schmitt-NAND gate(both TTL and MOS)
- Study of Arithmetic Logic Unit

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination
Examination Scheme:

Components	Quiz	Attendance	Mid Term Exam	Presentation/ Assignment/ etc.	End Term Exam
Weightage (%)	10	10	20	10	50

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Identify relevant information to supplement to the Digital Electronic ETEC210A course	**PSO1, PSO3**
CO2	Construct basic combinational circuits and verify their functionalities	**PO2**
CO3	To understand the basic digital circuits and to verify their operation.	**PO1**
CO4	To understand the concepts of flipflops, registers and counters.	**PO4**

CO5	To understand how gates are the basic building blocks for digital world.	**PO3**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2	PSO3
ETEC256A	Digital Electronics Lab	2	2	3	2									1		3

1=weakly mapped
2= moderately mapped
3=strongly mapped

Semester IV

ETCS222A	Computer Organization and Architecture	L	T	P	C
Version 1.0		3	1	-	4
Pre-requisites/Exposure	Basics of Microprocessor Systems				
Co-requisites	-				

Course Objectives

1. How Computer Systems work & the basic principles?
2. Instruction Level Architecture and Instruction Execution
3. The current state of art in memory system design
4. How I/O devices are accessed and its principles?
5. To provide the knowledge on Instruction Level Parallelism
6. To impart the knowledge on micro programming
7. Concepts of advanced pipelining techniques.

Course Outcomes

On completion of this course, the students will be able to

CO1. Understand the concepts of microprocessors, their principles and practices.

CO2. Write efficient programs in assembly language of the 8086 family of microprocessors.

CO3. Organize a modern computer system and be able to relate it to real examples.

CO4. Develop the programs in assembly language for 80286, 80386 and MIPS processors in real and protected modes.

CO5. Implement embedded applications using Emulator.

Catalog Description

Computer architecture is the science and art of selecting and interconnecting hardware components to create a computer that meets functional, performance, and cost goals. Computer organization defines the constituent parts of the system, how they are interconnected, and how they interoperate in order to implement the architectural specification. In this course, you will learn the basics of hardware components from basic gates to memory and I/O devices, instruction set architectures and assembly language, and designs to improve performance.

Course Content

Unit I: **12 lecture hours**

Functional blocks of a computer: CPU, memory, input-output subsystems, control unit. Instruction set architecture of a CPU–registers, instruction execution cycle, RTL interpretation of instructions, addressing modes, instruction set. Case study – instruction sets of some common CPUs.

Data representation: signed number representation, fixed and floating point representations, character representation. Computer arithmetic – integer addition and subtraction, ripple carry adder, carry look- ahead adder, etc. multiplication – shift- and add, Booth multiplier, carry save multiplier, etc. Division restoring and non-restoring techniques, floating point arithmetic.

Unit II: **10 lecture hours**

Introduction to x86 architecture.

CPU control unit design: hardwired and micro-programmed design approaches, Case study – design of a simple hypothetical CPU.

Memory system design: semiconductor memory technologies, memory organization.

Peripheral devices and their characteristics: Input-output subsystems, I/O device interface, I/O transfers–program controlled, interrupt driven and DMA, privileged and non-privileged instructions, software interrupts and exceptions. Programs and processes–role of interrupts in process state transitions, I/O device interfaces – SCII, USB

Unit III: **8 lecture hours**

Pipelining: Basic concepts of pipelining, throughput and speedup, pipeline hazards.

Parallel Processors: Introduction to parallel processors, Concurrent access to memory andcache coherency.

Unit IV: **10 lecture hours**

Memory organization: Memory interleaving, concept of hierarchical memory organization, cache memory, cache size vs. block size, mapping functions, replacement algorithms, write policies.

Text Books

1. "Computer Organization and Design: The Hardware/Software Interface", 5th Edition by David A. Patterson and John L. Hennessy, Elsevier.

2. "Computer Organization and Embedded Systems", 6th Edition by Carl Hamacher, McGraw Hill Higher Education.

Reference Books/Materials

1. "Computer Organization and Design: The Hardware/Software Interface", 5th Edition by David A. Patterson and John L. Hennessy, Elsevier.

2. "Computer Organization and Embedded Systems", 6th Edition by Carl Hamacher, McGraw Hill Higher Education.

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination

Examination Scheme:

Components	Quiz	Attendance	Mid Term Exam	Presentation/ Assignment/ etc.	End Term Exam
Weightage (%)	10	10	20	10	50

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	Mapped Program Outcomes
CO1	Understand the concepts of microprocessors, their principles and practices.	PO2
CO2	Write efficient programs in assembly language of the 8086 family of microprocessors.	PO3
CO3	Organize a modern computer system and be able to relate it to real examples.	PO4
CO4	Develop the programs in assembly language for 80286, 80386 and MIPS processors in real and protected modes.	PO9
CO5	Implement embedded applications using Emulator.	PO5

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2	PSO3
ETCS 222A	Computer Organization and Architecture		2	3	3	2				3						3

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCS220A	Analysis and Design of Algorithms	L	T	P	C
Version 1.0		3	1	0	4
Pre-requisites/Exposure	Advanced Computer Programming				
Co-requisites	--				

Course Objectives

1. The student should be able to choose appropriate data structures, understand the ADT/libraries, and use it to design algorithms for a specific problem.
2. Students should be able to understand the necessary divide and conquer algorithms.
3. To familiarize students with greedy and dynamic programming concepts
4. Student should be able to come up with analysis of efficiency and proofs of correctness.

Course Outcomes

On completion of this course, the students will be able to
CO 1 Analyze the asymptotic performance of algorithms.
CO 2 Write rigorous correctness proofs for algorithms.
CO 3 Demonstrate a familiarity with major algorithms and data structures.
CO 4 Apply important algorithmic design paradigms and methods of analysis.
CO 5 Synthesize efficient algorithms in common engineering design situations.

Catalog Description

This course introduces basic methods for the design and analysis of efficient algorithms emphasizing methods useful in practice. Different algorithms for a given computational task are presented and their relative merits evaluated based on performance measures. The following important computational problems will be discussed: sorting, searching, elements of dynamic programming and greedy algorithms, advanced data structures, graph algorithms (shortest path, spanning trees, tree traversals), string matching, elements of computational geometry, NP completeness

Course Content

Unit I: **8 lecture hours**

Introduction: Characteristics of algorithm. Analysis of algorithm: Asymptotic analysis of complexity bounds – best, average and worst-case behavior; Performance measurements of Algorithm, Time and space trade- offs, Analysis of recursive algorithms through recurrence relations: Substitution method, Recursion tree method and Masters' theorem.

Unit II: **12 lecture hours**

Fundamental Algorithmic Strategies: Brute -Force, Greedy, Dynamic Programming, Branch-and-Bound and Backtracking methodologies for the design of algorithms; Illustrations of these techniques for Problem-Solving, Bin Packing, Knap Sack TSP. Heuristics – characteristics and their application domains.

Unit III: **12 lecture hours**

Graph and Tree Algorithms: Traversal algorithms: Depth First Search (DFS) and Breadth First Search (BFS); Shortest path algorithms, Transitive closure, Minimum Spanning Tree, Topological sorting, Network Flow Algorithm.

Unit IV: **8 lecture hours**

Tractable and Intractable Problems: Computability of Algorithms, Computability classes – P, NP, NP-complete and NP-hard. Cook's theorem, Standard NP-complete problems and Reduction techniques. Advanced Topics: Approximation algorithms, Randomized algorithms, Class of problems beyond NP – P SPACE

Text Books

1. Introduction to Algorithms, 4TH Edition, Thomas H Cormen, Charles E Lieserson, Ronald L Rivest and Clifford Stein, MIT Press/McGraw-Hill.
2. Fundamentals of Algorithms – E. Horowitz et al.

Reference Books/Materials

1. Schaum's outline series, "Data Structure", McGraw Hills.

2. Y. Langsamet. al., "Data Structures using C and C++", PHI.

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination
Examination Scheme:

Components	Quiz	Attendance	Mid Term Exam	Presentation/ Assignment/ etc.	End Term Exam
Weightage (%)	10	10	20	10	50

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Analyze the asymptotic performance of algorithms.	**PO1**
CO2	Write rigorous correctness proofs for algorithms.	**PO4**
CO3	Demonstrate a familiarity with major algorithms and data structures.	**PO5**
CO4	Apply important algorithmic design paradigms and methods of analysis.	**PO2**
CO5	Synthesize efficient algorithms in common engineering design situations.	**PSO3**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
ETCS 220A	Analysis and design of algorit	2	2		3	3										3

	hms															

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETEC202A	**Signals & Systems**	L	T	P	C
Version 1.0		3	1	0	4
Pre-requisites/Exposure					
Co-requisites	--				

Course Objective:

- Be able to describe signals mathematically and understand how to perform mathematical operations on signals
- Be familiar with commonly used signals such as the unit step, ramp, impulse function, sinusoidal signals and complex exponentials.
- Be able to classify signals as continuous-time vs. discrete-time, periodic vs. non-periodic, energy signal vs. power signal, odd vs. even, conjugate symmetric vs anti-symmetric
- Be able to compute the output of an LTI system given the input and the impulse response through convolution sum and convolution integral.

Course Outcomes:

CO1 Represent and classify various types of signals and systems.

CO2 Analyze the spectral characteristics of continuous-time and discrete time signals using Fourier analysis and will be able to find Fourier transform for different signals.

CO3 Classify systems based on their properties and determine the response of LTI systems.

CO4 Analyze the system properties based on impulse response and Fourier analysis.

CO5 Apply the Laplace transform and Z-Transform for analyse of continuous time and discrete time signals and systems.

CO6 Understand the process of sampling and the effects of under sampling.

Catalog Description:

The objective of the course is to provide brief methodologies for analysis of Signals and Systems to the engineering students. The course module includes introduction of signals and their elementary operations, Laplace and Fourier analysis, Systems and their analysis and Z-Transform.

Course Content

UNIT – I **10 Lecture Hours**

Types of signals and Elementary operations: Signal Classification: Deterministic and Stochastic, discrete and continuous signals, analog and digital signals, periodic and a periodic, energy and power signals, causal and non-causal signals, one dimensional and multidimensional signals etc., impulse functional sequences, analog and discrete, singular functions. Signal representation in terms of singular functions, orthogonal functions and their use in signal representation.

UNIT-II **10 Lecture Hours**

Laplace and Fourier analysis: Fourier series, Fourier and Laplace transforms: properties and applications, Signal characterization using fourier and Laplace transform, Convolution theorem: geometrical interpretation and applications.

Discretization of Analog Signals: sampling, sampling theorem and its proof. Effect of under Sampling, recovery of analog signals from sampled signal: reconstruction formula.

UNIT-III **12 Lecture Hours**

Z-Transform: Introduction and properties of Z-transform, Methods of Z-inversion: Inverse Z-transform by Partial fraction, long-division method and C-R Theorem, Applications of Z-transform. System Classification: linear and non-linear, time invariant and time varying, lumped and distributed, Deterministic and Stochastic. Casual and non-causal, Analog and Discrete/Digital, memory and memory less, 1 port and N – port, SISO, SIMO, MISO, MIMO.

UNIT-IV **12 Lecture Hours**

System Modeling: System Models in terms of differential, equations, state variables, difference equations and transfer functions. System Analysis: Linear time invariant system properties, elementary idea of response determination to deterministic and stochastic signals. Elementary concept of impulse response.

Text books

- Simon Haykins – “Signal & Systems”, Wiley Eastern

Reference books

- I J NAGRATH, R. RANJAN, "Signal and Systems", TMH, New Delhi.
- Simon Haykin& Barry Van Veen, " Signals and Systems", John Wiley &Son.
- A.V.Oppenheim, A.S.Willsky&A. Nawab, "Signals and Systems" Pearson Education.

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination Scheme:

Components	Quiz	Attendance	Mid Term Exam	Presentation/ Assignment	End Term Exam
Weightage (%)	10	10	20	10	50

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and Pos		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Represent and classify various types of signals and systems.	**PSO1**
CO2	Analyze the spectral characteristics of continuous-time and discrete time signals using	**PO4, PSO3**
CO3	Classify systems based on their properties and determine the response of LTI systems.	**PO1, PSO3**
CO4	Analyze the system properties based on impulse response and Fourier analysis.	**PO4**
CO5	Apply the Laplace transform and Z-Transform for analyses of continuous time.	**PO1**
CO6	Understand the process of sampling and the effects of under sampling.	**PO2**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
ETEC 202A	SIGNAL & SYSTEMS	2	2		3									3		3

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCS 202A	Software Engineering	L	T	P	C
Version 1.0		3	1	0	4
Pre-requisites/Exposure	None				
Co-requisites	--				

Course Objectives

1. The aim of the course is to provide an understanding of the working knowledge of the techniques for estimation, design, testing and quality management of large software development projects.
2. Topics include process models, software requirements, software design, software testing, software process/product metrics, risk management, quality management and UML diagrams

Course Outcomes

On completion of this course, the students will be able to:

CO1. To learn and understand the Concepts of Software Engineering
CO2. To Learn and understand Software Development Life Cycle
CO3. To apply the project management and analysis principles to software project development.
CO4. To apply the design & testing principles to software project development.
CO5. Ability to execute tests, design test cases, use test tools, etc.
CO6. To Study about Software maintenance tools

Catalog Description

This course covers the fundamentals of software engineering, including understanding system requirements, finding appropriate engineering compromises, effective methods of design, coding, and testing, team software development, and the application of engineering tools.

Course Content

Unit I: **10 lecture hours**

Introduction: Software Crisis, Software Processes & Characteristics, Software life cycle models, Waterfall, Prototype, Evolutionary and Spiral Models

Software Requirements analysis & specifications: Requirement engineering, requirement elicitation techniques, requirements analysis using DFD, Data dictionaries & ER Diagrams, Requirement documentation, Nature of SRS, Characteristics & organization of SRS.

Unit II: **12 lecture hours**

Software Metrics: Software measurements: What & Why, Token Count, Size Estimation like lines of Code & Function Count, Halstead Software Science Measures, Design Metrics, Data Structure Metrics, Information Flow Metrics, Cost Estimation Models: COCOMO,COCOMO-II.

System Design: Design Concepts, design models for architecture, component, data and user interfaces; Problem Partitioning, Abstraction, Cohesiveness, Coupling, Top Down and Bottom-Up design approaches; Functional Versus Object Oriented Approach, Design Specification.

Coding: TOP-DOWN and BOTTOM-UP structure programming, Information Hiding, Programming Style, and Internal Documentation, Verification.

Unit III: **11 lecture hours**

Unified Approach and Unified Modeling Language: The Unified Approach: Layered Approach to OO Software Development, UML: UML Diagrams for Structure Modeling, UML Diagrams for Behavior Modeling, UML Diagram for Implementation and deployment modeling.

Software Reliability: Importance, Hardware Reliability & Software Reliability, Failure and Faults, Reliability Models, Basic Model, Logarithmic Poisson Model, Software Quality Models, CMM & ISO 9001.

Unit IV: **12 lecture hours**

Software Testing: Testing process, Design of test cases, functional testing: Boundary value analysis, Equivalence class testing, Decision table testing, Cause effect graphing, Structural testing, Path Testing, Data flow and mutation testing, Unit Testing, Integration and System Testing, Debugging, Alpha & Beta Testing, Testing Tools & Standards.

Software Maintenance: Management of Maintenance, Maintenance Process, Maintenance Models, Regression Testing, Reverse Engineering, Software Re-engineering, Configuration Management, Documentation.

Text Books

1. K. K. Aggarwal & Yogesh Singh, “Software Engineering”, New Age International.
2. R. S. Pressman, “Software Engineering – A practitioner’s approach”, McGraw Hill Int. Ed.
3. W.S. Jawadekar, “Software Engineering – Principles and Practices”, McGraw Hill

Reference Books/Materials

1. Stephen R. Schach, "Classical & Object Oriented Software Engineering", IRWIN, TMH.
2. James Peter, W. Pedrycz, "Software Engineering: An Engineering Approach", John Wiley & Sons.
3. I. Sommerville, "Software Engineering", Addison Wesley.
4. K. Chandrasehakhar, "Software Engineering & Quality Assurance", BPB.

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination
Examination Scheme:

Components	Quiz	Attendance	Mid Term Exam	Presentation/ Assignment/ etc.	End Term Exam
Weightage (%)	10	10	20	10	50

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	Mapped Program Outcomes
CO1	To learn and understand the Concepts of Software Engineering	PO1
CO2	To Learn and understand Software Development Life Cycle	PO1
CO3	To apply the project management and analysis principles to software project development.	PO3, PO11
CO4	To apply the design & testing principles to software project development.	PO3
CO5	Ability to execute tests, design test cases, use test tools, etc.	PO4
CO6	To Study about Software maintenance tools	PO2, PO5

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behavior	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO4	PO 5	PO 6	PO7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2	PSO3
ETCS 202A	Software Engineering	3	3	3	3	3						2		3	2	3

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCA228A	**Mobile Application Development**	L	T	P	C
Version 1.0		4	0	0	4
Pre-requisites/Exposure	Java Programming				
Co-requisites	--				

Course Objectives

This course facilitates classroom and laboratory learning, letting students develop competence

and confidence in android programming and understand the entire Android Apps Development Cycle, as well as it would also enable the students to independently create new Android Applications.

Course Outcomes

On completion of this course, the students will be able to

CO1. Explain functioning of different mobile technology

CO2. Demonstrate Android activities life cycle

CO3. Execute operations on GUI objects

CO4. Perform Event driven programming

CO5 Apply various techniques on working with menu

Catalog Description

The *Android* operating system (*OS*) has the highest *market share* worldwide on *mobile* devices. *Android* held 71.93 percent of the market. It is therefore necessary for students to know that how to build mobile apps for android operating system. This course covers the necessary concepts which are required to understand mobile communication and to develop Android Applications.

Course Content

Unit I: 10 lecture hours

Introduction to Mobile Computing: Concept of Mobile Communication, Different generations of wireless technology, Basics of cell, cluster and frequency reuse concept, Noise and its effects on mobile, Understanding GSM and CDMA, Basics of GSM architecture and services like voice call, SMS, MMS, LBS, VAS, Different modes used for Mobile Communication, Architecture of Mobile Computing(3 tier), Design considerations for mobile computing, Characteristics of Mobile Communication, Application of Mobile Communication, Security Concern Related to Mobile Computing, Middleware and Gateway required for mobile Computing, Making Existing Application Mobile Enable, Mobile IP, Basic Mobile Computing Protocol

Unit II: 9 lecture hours

Introduction to Android Programming: Overview of Android, Android Internals, Android for mobile apps development, Environment setup for Android apps Development, Framework - Android-SDK, Emulators - Android AVD, Android Emulation – Creation and set up, First Android Application

Unit III: **9 lecture hours**

Android Activities and GUI Design: Activity Lifecycle of Android, Design criteria for Android Application : Hardware Design Consideration, Design Demands For Android application, Intent, Activity, Activity Lifecycle and Manifest, Creating Application and new Activities, Simple UI - Layouts and Layout properties: Introduction to Android UI Design, Introducing Layouts, Fragments, Push Button , Text / Labels , Edit Text, Toggle Button , Padding

Unit IV: **12 lecture hours**

Background Tasks: Customizations: Floating hints and Auto Complete, Create Custom Layout, Create Custom Toast.

Save Data Locally on Phone: Save User Preferences, Save data using text files, Making use of AsyncTask class: Intro to AsyncTaskLoader, load InBackground() , AsyncTask Loader callbacks , Benefits of loaders . Connecting to data by SQL Lite Database: Overview of SQLite, OpenHelper Android class, Querying (dev) Searching (user) databases, Best practices for using databases in Android, Best practices for testing your database

Permissions: The permissions model, Libraries: Using libraries, Widgets: What are widgets?, When to use them and how to implement them, Publishing your App: Different ways to monetize your app, Making and publishing APKs: Guidelines for publishing in Google Play , Make and sign the APK, Beta test your app , Publish your app to Google Play

Text Books

1. Reto Meier, "Professional Android Application Development", Wiley India Pvt Ltd
2. Mark L Murphy, "Beginning Android", Wiley India Pvt Ltd
3. Sayed Y Hashimi and SatyaKomatineni, "Pro Android", Wiley India Pvt Ltd

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination
Examination Scheme:

Components	Quiz	Attendance	Mid Term Exam	Presentation/ Assignment/ etc.	End Term Exam
Weightage (%)	10	10	20	10	50

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs

	Course Outcomes (COs)	Mapped Program Outcomes
CO1	Explain functioning of different mobile technology	**PO1**
CO2	Demonstrate Android Activities Life Cycles	**PO2**
CO3	Execute Operations on GUI objects	**PO3, PO5**
CO4	Perform Event Driven Programming	**PO5, PO9**
CO5	Apply various techniques on working with menu	**PO6, PO11, PO12**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3

ETCA228A	Mobile Application Development	1	2	3		3	2			2		2	2	3		2

1=weakly mapped;
2= moderately mapped;
3=strongly mapped

ETCS 260A	**Computer Organization & Architecture Lab**	L	T	P	C
Version 1.0		0	0	2	1
Pre-requisites/Exposure	Basics of Microprocessor Systems				
Co-requisites	--				

Course Objectives

1. Develop and assemble assembly programs.
2. Identify and use proper assembler directives.
3. Design simple assembly programs.
4. Write programs that interface with a programming language.
5. Appreciate the System Software development environment.

Course Outcomes

On completion of this course, the students will be able to

CO1. Understand the concepts of microprocessors, their principles and practices.

CO2. Write efficient programs in assembly language of the 8086 family of microprocessors.

CO3. Organize a modern computer system and be able to relate it to real examples.

CO4. Develop the programs in assembly language for 80286, 80386 and MIPS processors in real and protected modes.

CO5. Implement embedded applications using Emulator.

Catalog Description

This course is an attempt to familiarize students with some of the important Assemblers available in the Windows environment. Students may use any of these tools available. Students may also find that assembler directives used by these programs may differ. Assembly and C Programming helps students greatly in System Software implementation and giving understanding of the machine.

List of Experiments (Indicative)

1	Design and simulate ripple carry adders	2 lab hours
2	Design and simulate carry look ahead adders	2 lab hours
3	Design and simulate Wallace tree adders	2 lab hours
4	Synthesis of various flip-flops.	2 lab hours
5	Design and simulate various registers and counters	2 lab hours
6	Design and simulate combinational multipliers	3 lab hours
7	Design and simulate Booth's Multiplication	3 lab hours
8	Design and simulate arithmetic logic unit	3 lab hours
9	Design memory units and understand how it operates during read and write operation.	4 lab hours
10	Designing an associative cache for given parameters.	3 lab hours
11	Design a CPU to show the basic top-level functionality, organization and architecture of a computer.	4 lab hours

Modes of Evaluation: Quiz/Oral practical oral exam/presentation/projects/Practical Examination

Examination Scheme:

Components	**Quiz**	**Attendance**	**Mid Term**	**Presentation/**	**End Term**

			Exam	Assignment/ etc.	Exam
Weightage (%)	**10**	**10**	**20**	**10**	**50**

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Understand the concepts of microprocessors, their principles and practices.	**PO2**
CO2	Write efficient programs in assembly language of the 8086 family of microprocessors.	**PO3**
CO3	Organize a modern computer system and be able to relate it to real examples.	**PO4**
CO4	Develop the programs in assembly language for 80286, 80386 and MIPS processors in real and protected modes.	**PO9**
CO5	Implement embedded applications using Emulator.	**PO5**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge

Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
ETCS 260A	**Computer organization & architecture lab**		2	3	3	2				3				3		3

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCS262A	**Analysisand Designof Algorithms Lab**	L	T	P	C
Version 1.0		0	0	2	1
Pre-requisites/Exposure	Practical learning				
Co-requisites	--				

Course Objectives

1. To understand concept of different sorting algorithms.
2. To understand the concept of dynamic programming.
3. To understand concept of divide and conquer.
4. To understand Dictionary (ADT)
5. To understand concept of greedy algorithms.
6. To understand concept & features like max heap, min heap

Course Outcomes

On completion of this course, the students will be able to

CO 1 Student will be able to implement optimal solution for various dynamic problems.

CO 2 To understand various sorting techniques.

CO 3 Analyze working of various operations on graphs.

CO 4 To understand concept of string matching in data structure

Course Content

List of Experiments

1	To analyze time complexity of insertion sort	**2 lab hours**
2	To analyze time complexity of Quick sort	**2 lab hours**
3	To analyze time complexity of merge sort	**2 lab hours**
4	Implement Largest Common Subsequence.	**2 lab hours**
5	To Implement Optimal Binary Search Tree.	**2 lab hours**
6	To Implement Matrix Chain Multiplication.	**2 lab hours**
7	To Implement Strassen's matrix multiplication Algorithm.	**2 lab hours**
8	To implement Knapsack Problem.	**2 lab hours**
9	To implement Activity Selection Problem.	**2 lab hours**
10	To implement Dijkstra's Algorithm.	**2 lab hours**
11	To implement Warshall's Algorithm.	**2 Labs**
12	To implement Bellman Ford's Algorithm.	**2 Labs**
13	To implement Depth First Search Algorithm.	**1 Lab**

14	To implement Breadth First Search Algorithm.	**1 Lab**
15	To implement NaïveString MatchingAlgorithm.	**1 Lab**
16	To implement Rabin Karp String MatchingAlgorithm	**1 Lab**

Modes of Evaluation: Quiz/Oral practical oral exam/presentation/projects/Practical Examination

Examination Scheme:

Components	**Quiz**	**Attendance**	**Mid Term Exam**	**Presentation/ Assignment/ etc.**	**End Term Exam**
Weightage (%)	**10**	**10**	**20**	**10**	**50**

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Student able to implement program for graph representation.	**PO2**
CO2	To understand operations like insert and search record in the database.	**PO3**
CO3	Analyze working of various operations on AVL Tree.	**PO5**
CO 4	To understand concept of file organization in data structure	**PSO1, PO9**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12	PSO 1	PSO 2	PSO 3
ETCS262A	Analysis and design of algorithms Lab		2	3		3				3				3		

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCS 252A	Software Engineering Lab	L	T	P	C
Version 1.0		0	0	2	1
Pre-requisites/Exposure	None				
Co-requisites	--				

Course Objectives

1. The aim of the course is to provide an understanding of the working knowledge of the techniques for estimation, design, testing and quality management of large software development projects.
2. Topics include process models, software requirements, software design, software testing, software process/product metrics, risk management, quality management and UML diagrams

Course Outcomes

On completion of this course, the students will be able to:

CO1. To learn and understand the Concepts of Software Engineering
CO2. To Learn and understand Software Development Life Cycle
CO3. To apply the project management and analysis principles to software project development.
CO4. To apply the design & testing principles to software project development.
CO5. Ability to execute tests, design test cases, use test tools, etc.
CO6. To Study about Software maintenance tools

Catalog Description

Based on theory subject **ETCS 202A**, the following experiments are to be performed. It enables students to understand the Software Engineering concept and use them practically to develop quality software.

List of Experiments (Indicative)

1	To identify the role of the software in today's world across a few significant domains related to day-to-day life Create SRS document of admission management for your university	**2 lab hours**
2	To identify the problem related to software crisis for a given scenario	**2 lab hours**
3	To identify the suitable software development model for the given scenario.	**2 lab hours**

4	To identify the various requirement development activities viz. elicitation, analysis, specification and verification for the given scenario	**4 lab hours**
5	To identify the various elicitation techniques and their usage for the Banking case study.	**4 lab hours**
6	Identify the elements in Software Requirements Specification for a given document.	**2 lab hours**
7	Draw E-R Diagram for Hockey League.	**2 lab hours**
8	Draw a context diagram and a level-1 diagram that represent the selling system at the store**.**	**2 lab hours**
9	Find out all software metrics for a Quadratic Equation program written in 'C'.	**2 lab hours**
10	Identify the design principle that is being violated in relation to the given scenario.	**2 lab hours**
11	To identify the usage of stubs or drivers in the context of an integration testing scenario.	**2 lab hours**
12	Identify the different types of performance testing.	**2 lab hours**
13	Identify the usage of regression testing.	**2 lab hours**
14	Write various white box test cases to test the internal behaviour of above program.	**2 lab hours**
15	Write various Black box test cases to test the functionalities of above program.	**2 lab hours**

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination
Examination Scheme:

Components	**Quiz**	**Attendance**	**Mid Term Exam**	**Presentation/ Assignment/ etc.**	**End Term Exam**
Weightage (%)	**10**	**10**	**20**	**10**	**50**

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	To learn and understand the Concepts of Software Engineering	**PO1**
CO2	To Learn and understand Software Development Life Cycle	**PO1**
CO3	To apply the project management and analysis principles to software project development.	**PO3, PO11**
CO4	To apply the design & testing principles to software project development.	**PO3**
CO5	Ability to execute tests, design test cases, use test tools, etc.	**PO4**
CO6	To Study about Software maintenance tools	**PO2, PO5**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behavior	Knowledge

Course Code	Course Title	PO 1	PO 2	PO 3	PO4	PO 5	PO 6	PO7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2	PSO3
ETCS 252A	Software Engineering Lab	3	3	3	3	3						2		3	2	3

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCS264A	**Mobile Application Develpoment Lab**	L	T	P	C
Version 1.0		0	0	2	1
Pre-requisites/Exposure	Java Programming				
Co-requisites	--				

Course Objectives

This course facilitates classroom and laboratory learning, letting students develop competence and confidence in android programming and understand the entire Android Apps Development Cycle, as well as it would also enable the students to independently create new Android Applications.

Course Outcomes

On completion of this course, the students will be able to

CO1. Design User Interface and develop activity for Android App.

CO2. Use Intent, Broadcast receivers and Internet services in Android App.

CO3. Design and implement Database Application and Content providers.

CO4. Use multimedia, camera and Location based services in Android App

Catalog Description

This course complements ETCA264A. The course acquaints the students with various features of Android programming. The aim of the course is to organizing the data in variety of ways using data structures and solve the given problem efficiently. Java is primary language for developing applications.

List of Experiments (Indicative)

1	Getting Started with Android Development.	**2 lab hours**
2	Activities and Views: Android Manifest.xml, Activity Class, Basic View Components: Layouts and Buttons	**2 lab hours**
3	Navigation with Data: Working with Intent, Sharing Data between Activities, Application Class.	**4 lab hours**
4	Android Resources: String Resources, Loading Strings in XML, Loading Strings in Code, the Resource Values Folder	**2 lab hours**
5	Drawables - Image Basics, Drawable Folders and Qualifiers, Dimensions, Image Padding, The ImageButton Widget	**2 lab hours**
6	Lists Implementing an Android List, ListView, ListActivity, Empty Lists, ListAdapter, Sorting the Adapter, Overriding ArrayAdapter, List Interaction	**4 lab hours**
7	Dialogs, New and Old: AlertDialog, Custom Dialog, Support Library, Fragments, DialogFragment.	**2 lab hours**
8	Menus: Options Menu, Modifying an Options Menu, Context Menu	**3 lab hours**
9	Saving Data with Shared Preferences: Shared Preferences, Getting Started with Shared Preferences, Preference Activity	**4 lab hours**

10	Saving Data with a Database: Setting Up SQLite, Creating a Helper , using the Helper, Cursor and Cursor Adapater	**2 lab hours**
11	Threading with AsyncTasks: Threading in Android, AsyncTask, Tracking Progress	**2 lab hours**
12	Styles and Themes: Introduction to Styling: Defining Styles, Defining Themes, Style Inheritance, Direct Theme References	**2 lab hours**
13	Develop an Android based Project	**4 lab hours**

Modes of Evaluation: Quiz/Oral practical oral exam/presentation/projects/Practical Examination

Examination Scheme:

Components	**Quiz**	**Attendance**	**Mid Term Exam**	**Presentation/ Assignment/ etc.**	**End Term Exam**
Weightage (%)	**10**	**10**	**20**	**10**	**50**

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Design User Interface and develop activity for Android App.	**PO1; PO5**
CO2	Use Intent, Broadcast receivers and Internet services in Android App.	**PO2; PO3**
CO3	Design and implement Database Application and Content providers.	**PO3; PO9**
CO4	Use multimedia, camera and Location based services in Android App	**PO11; PO12**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2	PSO3
ETCS264A	Mobile Application Development Lab	2	2	3		3				2		3	2			3

1=weakly mapped
2= moderately mapped
3=strongly mappe

Semester V

ETCS214A	Theory of Computation	L	T	P	C
Version 1.0		3	1	0	4
Pre-requisites/Exposure	Discrete Mathematics				
Co-requisites	--				

Course Objectives

1. Develop a formal notation for strings, languages and machines.
2. Design finite automata to accept a set of strings of a language.
3. Prove that a given language is regular and apply the closure properties of languages.
4. Design context free grammars to generate strings from a context free language and convert them into normal forms.
5. Prove equivalence of languages accepted by Pushdown Automata and languages generated by context free grammars.
6. Identify the hierarchy of formal languages, grammars and machines.
7. Distinguish between computability and non-computability and Decidability and undecidability.

Course Outcomes

On completion of this course, the students will be able to

CO1. Write a formal notation for strings, languages and machines.
C02. Design finite automata to accept a set of strings of a language.
CO3. Determine equivalence of languages accepted by Pushdown Automata and languages generated by context free grammars.
CO4. Distinguish between computability and non-computability and Decidability and undecidability.

Catalog Description

This course provides a formal connection between algorithmic problem solving and the theory of languages and automata and develop them into a mathematical view towards algorithmic design and in general computation itself. The course should in addition clarify the practical view towards the applications of these ideas in the engineering part of computer science.

Course Content

Unit I: **12 lecture hours**

Introduction to formal proof: Additional forms of proof, Inductive proofs, Finite Automata (FA), Deterministic Finite Automata (DFA), Non-deterministic Finite Automata (NFA), Finite Automata with Epsilon transitions.

Unit II: **8 lecture hours**

Regular Expression: FA and Regular Expressions, Proving languages not to be regular, Closure properties of regular languages, Equivalence and minimization of Automata.

Unit III: **12 lecture hours**

Context-Free Grammar (CFG): Parse Trees, Ambiguity in grammars and languages, Definition of the Pushdown automata, Languages of a Pushdown Automata, Equivalence of Pushdown automata and CFG, Deterministic Pushdown Automata. Normal forms for CFG, Pumping Lemma for CFL, Closure Properties of CFL, Turing Machines, Programming Techniques for TM.

Unit IV: **8 lecture hours**

A language that is not Recursively Enumerable (RE): An undecidable problem that is RE, Undecidable problems about Turing Machine, Post's Correspondence Problem.

Text Books

1. J.E. Hopcroft, R. Motwani and J.D. Ullman, "Introduction to Automata Theory, Languages and Computations", second Edition, Pearson Education.

Reference Books/Materials

1. H.R. Lewis and C.H. Papadimitriou, "Elements of the theory of Computation", Second Edition, Pearson Education.

2. Thomas A. Sudkamp," An Introduction to the Theory of Computer Science, Languages and Machines", Third Edition, Pearson Education.

3. Raymond Greenlaw an H.James Hoover, "Fundamentals of Theory of Computation, Principles and Practice", Morgan Kaufmann Publishers.

4. MichealSipser, "Introduction of the Theory and Computation", Thomson Brokecole.

5. J. Martin, "Introduction to Languages and the Theory of computation" Third Edition, Tata

McGraw Hill.

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination
Examination Scheme:

Components	Quiz	Attendance	Mid Term Exam	Presentation/ Assignment/ etc.	End Term Exam
Weightage (%)	**10**	**10**	**20**	**10**	**50**

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Write a formal notation for strings, languages and machines	**PO1**
CO2	Design finite automata to accept a set of strings of a language	**PO3**
CO3	Determine equivalence of languages accepted by Pushdown Automata and languages generated by context free grammars	**PO2**
CO4	Distinguish between computability and non-computability and Decidability and un-decidability	**PO4**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge

Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
ETCS214 A	Theory of Computatio n	2	3	3	3									3		

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCS304A	**Computer Networks**	L	T	P	C
Version 1.0		3	1	0	4
Pre-requisites/Exposure	Basics of Data Structure and Algorithms				
Co-requisites	Basic Mathematics				

Course Objectives

1. Help in understanding the concepts of communication and computer networks.

Course Outcomes

On completion of this course, the students will be able to

CO1. To develop an understanding of modern network architectures from a design and performance perspective.
CO2. To introduce the student to the major concepts involved in wide-area networks (WANs), local area networks (LANs) and Wireless LANs (WLANs).
CO3. To provide an opportunity to do network programming
CO4. Explain the functions of the different layer of the OSI Protocol.
CO5. For a given requirement (small scale) of wide-area networks (WANs), local area networks (LANs) and Wireless LANs (WLANs) design it based on the market available component

Catalog Description

Through this subject, student will be able to understand the coarse grained aspects of Data Communication. Student will understand the applications of data structures and algorithms in networks. The internals of communications will be discussed throughout the course duration.

Course Content

Unit I: **8 lecture hours**

Data communication Components: Representation of data and its flow Networks , Various Connection Topology, Protocols and Standards, OSI model, Transmission Media, LAN: Wired LAN, Wireless LANs, Connecting LAN and Virtual LAN, Techniques for Bandwidth utilization: Multiplexing - Frequency division, Time division and Wave division, Concepts on spread spectrum

Unit II: **12 lecture hours**

Data Link Layer and Medium Access Sub Layer: Error Detection and Error Correction - Fundamentals, Block coding, Hamming Distance, CRC; Flow Control and Error control protocols - Stop and Wait, Go back – N ARQ, Selective Repeat ARQ, Sliding Window, Piggybacking, Random Access, Multiple access protocols -Pure ALOHA, Slotted ALOHA, CSMA/CD,CDMA/CA

Unit III: **12 lecture hours**

Network Layer: Switching, Logical addressing – IPV4, IPV6; Address mapping – ARP, RARP, BOOTP and DHCP–Delivery, Forwarding and Unicast Routing protocols.

Transport Layer: Process to Process Communication, User Datagram Protocol (UDP), Transmission Control Protocol (TCP), SCTP Congestion Control; Quality of Service, QoS improving techniques: Leaky Bucket and Token Bucket algorithm.

Unit IV: **8 lecture hours**

Application Layer: Domain Name Space (DNS), DDNS, TELNET, EMAIL, File Transfer Protocol (FTP), WWW, HTTP, SNMP, Bluetooth, Firewalls, Basic concepts of Cryptography

Text Books

1. Data Communication and Networking, 4th Edition, Behrouz A. Forouzan, McGraw-Hill.

2. Data and Computer Communication, 8th Edition, William Stallings, Pearson Prentice Hall India.

Reference Books/Materials

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination
Examination Scheme:

Components	Quiz	Attendance	Mid Term Exam	Presentation/ Assignment/ etc.	End Term Exam
Weightage (%)	10	10	20	10	50

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	To develop an understanding of modern network architectures from a design and performance perspective.	**PO2, PO12**
CO2	To introduce the student to the major concepts involved in wide-area networks (WANs), local area networks (LANs) and Wireless LANs (WLANs).	**PO12**
CO3	To provide an opportunity to do network programming	**PO2**
CO4	Explain the functions of the different layer of the OSI Protocol.	**PO4, PO5**
CO5	For a given requirement (small scale) of wide-area networks (WANs), local area networks (LANs) and Wireless LANs (WLANs) design it based on the market available component	**PO11, PO12**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
ETCS304 A	Computer Networks		3		3	3						3	3	2	2	

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCS 206A	Computer Graphics	L	T	P	C
Version 1.0		3	1	0	4
Pre-requisites/Exposure					
Co-requisites	--				

Course Objectives

1. Have knowledge and understanding of the structure of an interactive computer graphics system, and the separation of system components.
2. Have knowledge and understanding of geometrical transformations and 3D viewing.
3. Have knowledge and understanding of techniques for representing 3D geometrical objects.
4. Have knowledge and understanding of interaction techniques.

Course Outcomes

On completion of this course, the students will be able to

CO1. Acquire familiarity with the concepts and relevant mathematics of computer graphics.

CO2. Implement various algorithms to scan, convert the basic geometrical primitives.

CO3. Describe the importance of viewing and projections.

CO4. To design basic graphics application programs.

CO5. Be able to design applications that display graphic images to given specifications.

CO6. understand a typical graphics pipeline.

Course Overview:

This course aims at familiarizing the student with basic transformation techniques, Curves, Projections etc. The course contains various Clipping Algorithms. A focus will be put on knowledge of computer-based graphics creation so that at the student at end of the course is well equipped to pursue either an industrial or academic career in the area.

Course Content

Unit I: **8 lecture hours**

Transformation, Projections, and Clipping Algorithms: Introduction to computer graphics, applications, hardware and software, 2D graphics, Bresenham's Line Drawing Algorithm, Homogeneous Coordinate System for 2D and 3D, Various 2D, 3D Transformation matrices (Translation, Scaling, Rotation, Shear), Rotation about an arbitrary point (2D), Rotation about an arbitrary axis (3D), Computing location of V.P, Clipping Algorithms, Sutherland-Cohen Clipping Algorithm.

Unit II: 12 hours

Curves and Surfaces: Bresenham's Circle Drawing Algorithm, Bezier Curves, 4 point and 5 point Bezier curves using Bernstein Polynomials, Conditions for smoothly joining curve segments, Bezier bi-cubic surface patch, B-Spline Curves, Cubic B-Spline curves using uniform knot vectors, Testing for first and second order continuities

Unit III: 12 hours

Projection and Solid Modelling: Parallel Projection, Oblique Projection on xy plane, Isometric Projection, Perspective Projection, One Vanishing Point (V.P.) projection from a point on z axis, Generation of 2 V.P. Projection, Isometric Projection, Perspective, Projection, one vanishing Pint (VP), projection from 0 point on z axis, Generation of 2 VP Projector & Projections, Solid Modelling.

Unit IV: 8 hours

Shading and Hidden Surface Removal: Shading, Illumination Model for diffused Reflection, Effect of ambient lighting, distances, Specular Reflection Model, Computing Reflection Vector, Curved Surfaces, Polygonal Approximations, Gourard Shading, Phong Model, Hidden Surface Removal, Back Face Detection, Depth Buffer (Z-Buffer, A-Buffer) Method, Scan Line Method, Depth Sorting Method, Area Subdivision Method.

TEXT BOOKS:

1. Foley et. al., "Computer Graphics Principles & practice", Addison Wesley.

REFERENCES BOOKS:

1. D. Rogers and J. Adams, "Mathematical Elements for Computer Graphics", MacGraw-Hill International Edition.
2. D. Hearn and P. Baker, "Computer Graphics", Prentice Hall.
3. R. Plastock and G. Kalley, "Theory and Problems of Computer Graphics", Schaum's Series, McGraw Hill.

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination
Examination Scheme:

Components	Quiz	Attendance	Mid Term Exam	Presentation/ Assignment/ etc.	End Term Exam

Weightage (%)	**10**	**10**	**20**	**10**	**50**

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Acquire familiarity with the concepts and relevant mathematics of computer graphics.	**PO1**
CO2	Implement various algorithms to scan, convert the basic geometrical primitives.	**PO4**
CO3	Describe the importance of viewing and projections.	**PO5**
CO4	Design basic graphics application programs.	**PO2**
CO5	Be able to design applications that display graphic images to given specifications.	**PO3**
CO6	Understand a typical graphics pipeline.	**PO1**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behavior	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2	PSO3
ETCS 206A	Computer Graphics	2	2	2	3	3								3		

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCS365A	**Computer Networks Lab**	L	T	P	C
Version 1.0		0	0	2	1
Pre-requisites/Exposure	Basics of Computer Programming				
Co-requisites	--				

Course Objectives

1. Learn basic concepts of computer networking and acquire practical notions of protocols with the emphasis on TCP/IP.
2. Provides a practical approach to assemble Ethernet/Internet networking.
3. Understanding of the layered architecture and working of important protocols

Course Outcomes

On completion of this course, the students will be able to

CO1. Understand the structure and organization of computer networks; including the division into network layers, role of each layer, and relationships between the layers.
CO2. Execute and evaluate network administration commands and demonstrate their use in different network scenarios.
CO3. Demonstrate and measure different network scenarios and their performance behavior.
CO4. Design and setup an organization network using packet tracer.

Catalog Description

This course complements ETCS304A. It enables them to select and design network for solving real life problem with optimal solution(s). The list of experiments helps to understand details of component of network and protocol.

List of Experiments (Indicative)

1	Study of Network devices in detail	**2 lab hours**
2	Connect the computers in Local Area Network using packet tracer	**2 lab hours**
3	Implementation of Data Link Framing method - Character Count.	**2 lab hours**
4	Implementation of Data link framing method - Bit stuffing and Destuffing.	**2 lab hours**
5	Implementation of Error detection method - even and odd parity.	**2 lab hours**
6	Implementation of Error detection method - CRC Polynomials.	**2 lab hours**
7	Implementation of Data Link protocols - Unrestricted simplex	**2 lab hours**

	protocol	
8	Implementation of data link protocols - Stop and Wait protocol	**2 lab hours**
9	Implementation of routing algorithms - Dijkstra's algorithm	**2 lab hours**
10	Study of Network IP Addressing using packet tracer	**2 lab hours**
11	Design TCP client and server application to transfer file	**2 lab hours**
12	Design UDP client and server application to transfer file	**2 lab hours**
13	Working on Network Protocol Analyzer Tool (Ethereal/Wireshark)	**4 lab hours**
14	Working on NMAP Tool for Port scanning	**4 lab hours**

Modes of Evaluation: Quiz/Oral practical oral exam/presentation/projects/Practical Examination

Examination Scheme:

Components	**Quiz**	**Attendance**	**Mid Term Exam**	**Presentation/ Assignment/ etc.**	**End Term Exam**
Weightage (%)	**10**	**10**	**20**	**10**	**50**

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Understand the structure and organization of computer networks; including the division into network layers, role of each layer, and relationships between the layers.	**PO2**
CO2	Execute and evaluate network administration commands and demonstrate their use in different network scenarios.	**PO3**

CO3	Demonstrate and measure different network scenarios and their performance behavior.	**PO5**
CO4	Design and setup an organization network using packet tracer.	**PO8**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2	PSO3
ETCS365A	Computer Networks Lab		3	3		2			3					3	3	

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCS258A	Computer Graphics Lab	L	T	P	C
Version 1.0		0	0	2	1
Pre-requisites/Exposure					
Co-requisites	--				

Course Objectives

1. To implement of line drawing, circle drawing, polygon drawing, transformation of objects, scaling, viewing, and curve designing and modeling algorithm practically for graphics.
2. To generate alternate solution for an existing problem with computer graphics.
3. Understand practical fundamental of line drawing, circle drawing, polygon drawing and curve drawing.

Course Outcomes

On completion of this course, the students will be able to

CO1.Understand the practical implementation of modeling, rendering, viewing ofobjects in2D and 3D.
CO2. Get knowledge about clipping algorithms
CO3. Get knowledge about Curves algorithms.
CO4. Implement Isometric Projection

Catalog Description

This course will provide basic concepts of computer graphics including necessary mathematics and algorithms. Primary focus of this course will be to understand the basics of 2D/3D rendering. The course will also cover various aspects of the rendering pipeline and realistic image synthesis using raytracing.

List of Experiments (Indicative)

1	Write a program to draw a point on screen. Study of various built-in commands to draw basic objects on screen.	**2 lab hours**
2	Write a program to implement Bresenham's Line Drawing Algorithm.	**2 lab hours**
3	Write a program to implement various 2D, 3D Transformation matrices such as Translation, Scaling, Rotation, and Shear.	**2 lab hours**

4	Write a program to implement Sutherland-Cohen line Clipping Algorithm.	**2 lab hours**
5	Write a program to implement Bresenham's Circle Drawing Algorithm.	**2 lab hours**
6	Write a program to implement Bezier Curves.	**2 lab hours**
7	Write a program to implement B-Spline Curves.	**2 lab hours**
8	Write a program to implement various Projections of 2D objects.	**2 lab hours**
9	Write a program to implement various Projections of 3D objects.	**4 lab hours**
10	Write a program to implement Isometric Projection.	**4 lab hours**

Modes of Evaluation: Quiz/Oral practical oral exam/presentation/projects/Practical Examination

Examination Scheme:

Components	**Quiz**	**Attendance**	**Mid Term Exam**	**Presentation/ Assignment/ etc.**	**End Term Exam**
Weightage (%)	**10**	**10**	**20**	**10**	**50**

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Understand the practical implementation of modeling, rendering, viewing of objects in 2D and 3D.	**PO2**
CO2	Get knowledge about clipping algorithms	**PO3**
CO3	Get knowledge about Curves algorithms.	**PO5**
CO4	Implement Isometric Projection	**PO1**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behavior	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2	PSO3
ETCS258A	Computer Graphics Lab	2	3	3		2								3		

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCS317A	Big Data Analytics	L	T	P	C
Version 1.0		3	0	0	3
Pre-requisites/Exposure	Data Structures and Algorithms				
Co-requisites	Database Management Systems				

Course Objectives

1. Help in understanding the information "hidden" within the voluminous data to make future business decisions.

Course Outcomes

On completion of this course, the students will be able to

CO1. Ability to identify the characteristics of datasets and compare the trivial data and big data for various applications.
C02. Ability to select and implement machine learning techniques and computing environment that are suitable for the applications under consideration.
CO3. Ability to solve problems associated with batch learning and online learning, and the big data characteristics such as high dimensionality, dynamically growing data and in particular scalability issues.
CO4. Ability to understand and apply scaling up machine learning techniques and associated computing techniques and technologies.
CO5. Ability to recognize and implement various ways of selecting suitable model parameters for different machine learning techniques.
CO6. Ability to integrate machine learning libraries and mathematical and statistical tools

Catalog Description

Through this subject, student will be able to understand the coarse grained aspects of analyzing and extracting relevant information from the vast repository. Student will implement the concepts of data structures and algorithms and database management systems to make highly precise decision from the given data set. The internals of smart analysis will be discussed throughout the course duration.

Course Content

Unit I: 8 lecture hours

Introduction to Big Data: Big Data characteristics, types of Big Data, Traditional vs. Big Data

business approach, Case Study of Big Data Solutions.

Overview of Hadoop: Core Hadoop Components, Hadoop Ecosystem, Physical Architecture, Hadoop limitations

Unit II: **12 lecture hours**

NoSQL: NoSQL business drivers; NoSQL case studies; NoSQL data architecture patterns: Key-value stores, Graph stores, Column family (Bigtable) stores, Document stores, Variations of NoSQL architectural patterns; Using NoSQL to manage big data, Understanding the types of big data problems; Analyzing big data with a shared-nothing architecture; Choosing distribution models: master-slave versus peer-to-peer

Unit III: **12 lecture hours**

MapReduce and the New Software Stack: Distributed File Systems -- Physical Organization of Compute Nodes, Large Scale File-System Organization, The Map Tasks, Grouping by Key, The Reduce Tasks, Combiners, Details of MapReduce Execution, Coping With Node Failures.

Unit IV: **8 lecture hours**

Algorithms Using MapReduce: Matrix-Vector Multiplication by MapReduce, Relational-Algebra Operations, Computing Selections by MapReduce, Computing Projections by MapReduce, Union, Intersection, and Difference by MapReduce, Computing Natural Join by MapReduce, Grouping and Aggregation by MapReduce, Matrix Multiplication, Matrix Multiplication with One MapReduce Step

Text Books

1. Data Analytics Made Accessible, A.Maheshwari.
2. Hadoop The definite Guide. 3rd edition

Reference Books/Materials

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination
Examination Scheme:

Components	Quiz	Attendance	Mid Term Exam	Presentation/ Assignment/ etc.	End Term Exam
Weightage (%)	10	10	20	10	50

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Ability to identify the characteristics of datasets and compare the trivial data and big data for various applications	**PO2,PO12**
CO2	Ability to select and implement machine learning techniques and computing environment that are suitable for the applications under consideration	**PO3**
CO3	Ability to solve problems associated with batch learning and online learning, and the big data characteristics such as high dimensionality, dynamically growing data and in particular scalability issues.	**PO5**
CO4	Ability to understand and apply scaling up machine learning techniques and associated computing techniques and technologies.	**PO3, PO4, PO5**
CO5	Ability to recognize and implement various ways of selecting suitable model parameters for different machine learning techniques	**PO3, PO5**
CO6	Ability to integrate machine learning libraries and mathematical and statistical tools	**PO2, PO5, PO12**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behavior	Knowledge
Course Code	Course Title	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
ETCS317A	Big Data Analytics		3	3	2	3							3	2		

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCS364A	Big Data Analytics Lab	L	T	P	C
Version 1.0		0	0	2	1
Pre-requisites/Exposure	Data Structures and Algorithms;				
Co-requisites	Database Management Systems				

Course Objectives

Understanding of processing of huge data set over clustered system.

Course Outcomes

On completion of this course, the students will be able to

CO1. Identify Big Data and its Business Implications.
CO2. List the components of Hadoop and Hadoop Eco-System.
CO3. Access and Process Data on Distributed File System.
CO4. Manage Job Execution in Hadoop Environment.
CO5. Develop Big Data Solutions using Hadoop Eco System.

Catalog Description

This course complements ETCS518A. It enables them to keenly analyze to reach a point of solving problems with the help of fundamentals. The list of experiments help organizing the flow of understanding and learning to solve the given problem efficiently.

List of Experiments (Indicative)

1	Set up a pseudo-distributed, single-node Hadoop cluster backed by the Hadoop Distributed File System, running on Ubuntu Linux. After successful installation on one node, configuration of a multi-node Hadoop cluster (one master and multiple slaves).	**2 lab hours**
2	MapReduce application for word counting on Hadoop cluster	**2 lab hours**
3	Unstructured data into NoSQL data and do all operations such as NoSQL query with API.	**2 lab hours**
4	K-means clustering using map reduce	**2 lab hours**
5	Page Rank Computation	**2 lab hours**
6	Mahout machine learning library to facilitate the knowledge build up in big data analysis.	**4 lab hours**

7	Application of Recommendation Systems using Hadoop/mahout libraries	**4 lab hours**

Modes of Evaluation: Quiz/Oral practical oral exam/presentation/projects/Practical Examination

Examination Scheme:

Components	**Quiz**	**Attendance**	**Mid Term Exam**	**Presentation/ Assignment/ etc.**	**End Term Exam**
Weightage (%)	**10**	**10**	**20**	**10**	**50**

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Identify Big Data and its Business Implications.	**PO7**
CO2	List the components of Hadoop and Hadoop Eco-System.	**PO1, PO3**
CO3	Access and Process Data on Distributed File System.	**PO1, PO2**
CO4	Manage Job Execution in Hadoop Environment.	**PO5**
CO5	Develop Big Data Solutions using Hadoop Eco System.	**PO2, PO3, PO12**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behavior	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2	PSO3
ETCS364A	Big Data Analytics Lab	2	3	3		2		2					2	2		3

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCS409A	**Advanced Computer Networks**	L	T	P	C
Version 1.0		3	0	0	3
Pre-requisites/Exposure	Basics of Computer Networks				
Co-requisites	--				

Course Objectives

1. To understand the state of the art in network protocols, network architecture, and networked systems.
2. To develop a strong understanding of the core concepts of computer networks
3. To gain practice of reading the research papers and critically understanding the research of others
4. Describe how computer networks are organized with the concept of layered approach with general principles of data communication
5. Describe how signals are used to transfer data between nodes and implement a simple LAN with hubs, bridges and switches.
6. To understand how packets in the Internet are delivered.

Course Outcomes

On completion of this course, the students will be able to

CO1. Independently understand basic computer network technology.
CO2. Understand and explain Data Communications System and its components.
CO3. Identify the different types of network topologies and protocols.
CO4. Enumerate the layers of the OSI model and TCP/IP. Explain the function(s) of each layer.
CO5. Identify the different types of network devices and their functions within a network
CO6. Understand and building the skills of subnetting and routing mechanisms.
CO7. Familiarity with the basic protocols of computer networks, and how they can be used to assist in network design and implementation.
CO8. Introduce the student to advanced networking concepts, preparing the student for entry Advanced courses in computer networking.

Catalog Description

This is a graduate level course on computer networking and assumes a student has a basic understanding of computer networks concepts. This course is a topics based course which primarily covers topics from Internet Architecture, Internet Congestion Control, Software Defined Networking, Delay Tolerant Networks, Wireless Networking, Quality of Service& Traffic Engineering, Network Performance & Management, Overlay Networks and Network Applications . In addition, this course will cover recent proposals to improve network performance, functionality and scalability to meet emergent applications
Requirement.

Course Content

Unit I: **12 lecture hours**

Internet Design & Architecture: Overview of network building blocks, Network architecture and design principles, layers and protocols: Internet Layering, Functionality Implementation (like Recovery from crashes, security, reliability etc.) at lower layers vs. Higher layers, Internet design: Challenges and Solutions, Case Study of Future Internet Design Project: Named Data Networking(NDN) Traffic Management: Congestion control principles, TCP congestion control, Load Balancing using Multipath TCP, IProuting: Intra-domain (OSPF/RIP) and Inter-domain (BGP), Adaptive Routing, Multipath and QoS Routing, Traffic Engineering Principles; Route Optimization, TE Issues and Challenges: Robustness, TE Interactions, Interoperability, MPLS Routing, Intradomain Routing: Protocols Characteristics and Limitations; Achieving QoS/Traffic Engineering with IP Routing Protocols.

Unit II: 8 lecture hours

Software Defined Networks (SDNs): Software Defined Networking (SDN): Centralized and Distributed Control and Data Planes, SDN Architecture, SDN Controllers, OpenFlow: Protocol to Program the Networks, Network Programmability, Network Function Virtualization, SDN Frameworks, Use cases for traffic monitoring& classification, bandwidth scheduling and monitoring. Delay Tolerant Networks (DTNs): Delay Tolerant Network Architecture, DTN Routing Protocols: Taxonomy and Design, Replication Based Routing Protocols, Open Issues and Challenges, DTN Application(s): Message Dissemination in Vehicular Networks, Adhoc Network for Disaster Rescue Management, Multimedia Content Delivery Network

Unit III: 12 lecture hours

Overlay Networks Applications: Overlay Networks: Advantages and Challenges, Resilient Overlay Networks(RON), Lookup Problem inP2P Networks, ScalableP2P Lookup Service for Internet Applications, Chord Protocol, DNS and Naming System, DNS and CDN,HTTP and CDN Case Study: Akamai CDN, An overlay approach to decouple sender and receiver to generalize the Multicast, Anycast and Mobility, Mobile P2P Overlay Networks for DTNs: Challenges, Prophet Based Information Retrieval, Ad-hoc Storage Overlay System

Unit IV: 8 lecture hours

Wireless Networks: Wireless Networking: An Overview, TCP Performance Issues in Wireless Links: Problems and Solutions, Network Centered IP Mobility Solutions, Overview of Wireless Sensor Networks.

Text Books

1. Data Communication and Networking, 4th Edition, Behrouz A. Forouzan, McGrawHill.

2. Data and Computer Communication, 8th Edition, William Stallings, Pearson Prentice Hall India.

Reference Books/Materials

1. "Data and Computer Communication" by William Stallings
2. "Computer Networks" by Andrew S Tanenbaum
3. "Internetworking with TCP/IP, Volume 1" by Douglas Comer

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination
Examination Scheme:

Components	Quiz	Attendance	Mid Term Exam	Presentation/ Assignment/ etc.	End Term Exam
Weightage (%)	10	10	20	10	50

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Independently understand basic computer network technology.	**PO2**
CO2	Understand and explain Data Communications System and its components.	**PO3**
CO3	Identify the different types of network topologies and protocols.	**PO4**
CO4	Enumerate the layers of the OSI model and TCP/IP. Explain the function(s) of each layer.	**PO5**
CO5	Identify the different types of network devices and their functions within a network	**PO4**
CO6	Understand and building the skills of subnetting and routing mechanisms.	**PO4**
CO7	Familiarity with the basic protocols of computer networks, and how they can be used to assist in network design and implementation.	**PO9**
CO8	Introduce the student to advanced networking concepts, preparing the student for entry Advanced courses in computer networking	**PSO3**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2	PSO3
ETCS409A	ADVANCED COMPUTER NETWORKS		2	3	3	3				3						3

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCS452A	Advanced Computer Networks Lab	L	T	P	C
Version 1.0		0	0	2	1
Pre-requisites/Exposure	Basics of Computer Networks				
Co-requisites	--				

Course Objectives

1. To develop an understanding of computer networking basics.
2. To develop an understanding of different components of computer networks, various protocols, modern technologies and their applications.
3. To recognize the technological trends of Computer Networking.
4. To understand the key technological components of the Network
5. To understand the state of the art in network protocols, network architecture, and networked systems.

Course Outcomes

On completion of this course, the students will be able to

CO1. Describe the general principles of data communication and how computer networks are organized with the concept of layered approach.

CO2. Understand and explain the concept of Data Communication and networks, layered architecture and their applications.

CO3. Describe how packets in the Internet are delivered and analyze the contents in a given data link layer

CO4. Evaluate data communication link considering elementary concepts of data link layer protocols for error detection and correction

CO5 Describe what classless addressing scheme is with how routing protocols work.

CO6. Estimate the congestion control mechanism to improve quality of service of networking application

Catalog Description

This course will introduce students to the basic design principles on which today's Internet is based upon a long with the current and emerging research topics in computer networking area. In addition, this course will cover recent proposals to improve network performance, functionality and scalability to meet emergent applications requirement. The list of experiments help in

understanding different computer network techniques.

List of Experiments (Indicative)

1	Implement the CRC-12, CRC-16 in data link layer	**2 lab hours**
2	Implement the data link protocols: Unrestricted simplex protocol	**2 lab hours**
3	Implement of one-bit sliding window protocol.	**2 lab hours**
4	Implement Dijkstra's algorithm to compute the shortest path thru a graph.	**3 lab hours**
5	Implement the Token Bucket Congestion control algorithm.	**3 lab hours**
6	Implement the Leaky Bucket Congestion control algorithm	**3 lab hours**
7	The Experiments using Mininet for Software Defined Network a. Network Toplology creation and REST API introduction. b. Influencing flows via cURL commands. c. Create a network and run a simple performance test. d. Use "ovs-vsctl" command to directly control open v switch. e. Dynamically change the network parameters—link delay analysis. f. Dynamically change the forwarding rules. g. Mininet Random Topology Generator.	**4 lab hours**

8	The experimenets using NS-3 a. Create a simple topology of two nodes (Node1, Node2) separated by a point-to-point link. Setup a UdpClient on one Node1 and a UdpServer on Node2. Let it be of a fixed data rate Rate1. Start the client application, and measure end to end throughput whilst varying the latency of the link. Now add another client application to Node1 and a server instance to Node2. What do you need to configure to ensure that there is no conflict? Repeat step 3 with the extra client and server application instances. Show screenshots of pcap traces which indicate that delivery is made to the appropriate server instance. b. Create a simple dumbbell topology, two client Node1 and Node2 on the left side of the dumbbell and server nodes Node3 and Node4 on the right side of the dumbbell. Let Node5 and Node6 form the bridge of the dumbbell. Use point to point links. Install a TCP socket instance on Node1 that will connect to Node3. Install a UDP socket instance on Node2 that will connect to Node4. Start the TCP application at time 1s. Start the UDP application at time 20s at rate Rate1 such that it clogs half the dumbbell bridge's link capacity. Increase the UDP application's rate at time 30s to rate Rate2 such that it clogs the whole of the dumbbell bridge's capacity. Use the ns-3 tracing mechanism to record changes in congestion window size of the TCP instance over time. Use gnuplot/matplotlib to visualise plots of cwnd vs time. Mark points of fast recovery and slow start in the graphs. Perform the above experiment for TCP variants Tahoe, Reno and New Reno, all of which are available with ns-3. c. Create a wireless mobile ad-hoc network with three nodes Node1, Node2 and Node3. Install the OLSR routing protocol on these nodes. Place them such that Node1 and Node3 are just out of reach of each other. Create a UDP client on Node1 and the corresponding server on Node3. Schedule Node1 to begin sending packets to Node3 at time 1s. Verify whether Node1 is able to send packets to Node3. Make Node2 move between Node1 and Node3 such that Node2 is visible to both A and C. This should happen at time 20s. Ensure that Node2 stays in that position for another 15s. Verify whether Node1 is able to send packets to Node3. At time	**8 lab hours**

	35s, move Node2 out of the region between Node1 and Node3 such that it is out of each other's transmission ranges again. Verify whether Node1 is able to send packets to Node3. To verify whether data transmissions occur in the above scenarios, use either the tracing mechanism or a RecvCallback() for Node3's socket. Plot the number of bytes received versus time at Node3. Show the pcap traces at Node 2's Wifi interface, and indicate the correlation between Node2's packet reception timeline and Node2's mobility.	

Modes of Evaluation: Quiz/Oral practical oral exam/presentation/projects/Practical Examination

Examination Scheme:

Components	Quiz	Attendance	Mid Term Exam	Presentation/ Assignment/ etc.	End Term Exam
Weightage (%)	10	10	20	10	50

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Describe the general principles of data communication and how computer networks are organized with the concept of layered approach.	**PO1**
CO2	Understand and explain the concept of Data Communication and networks, layered architecture and their applications.	**PO4**
CO3	Describe how packets in the Internet are delivered and analyze the contents in a given data link layer	**PO5**
CO4	Evaluate data communication link considering elementary concepts of data link layer protocols for error detection and correction	**PO2**
CO5	Describe what classless addressing scheme is with how routing protocols work.	**PO3**
CO6	Estimate the congestion control mechanism to improve quality of service of networking application	**PO6**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2	PSO3
ETCS452A	ADVANCED COMPUTER NETWORKS LAB	2	2	2	3	3	3									3

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCS410A	Mobile And Wireless Communication	L	T	P	C
Version 1.0		3	0	0	3
Pre-requisites/Exposure	Advanced of Computer communication				
Co-requisites	--				

Course Objectives

1. Define mobile technologies in terms of hardware, software, and communications.
2. Utilize mobile computing nomenclature to describe and analyze existing mobile computing frameworks and architectures.
3. Evaluate the effectiveness of different mobile computing frameworks.
4. Describe how mobile technology functions to enable other computing technologies.

Course Outcomes

On completion of this course, the students will be able to

CO1. Utilize mobile computing nomenclature to describe and analyze existing mobile computing frameworks and architectures.
CO2. Evaluate the effectiveness of different mobile computing frameworks.
CO3. Describe how mobile technology functions to enable other computing technologies.

Catalog Description

This course will cover the nomenclature and implementation of mobile computing and mobile communication. Coverage mobile systems will include 2G, 2.5G, 3G, 3G+, and 4G communication systems, mobile satellite communication networks, mobile IP, mobile TCP, digital audio-video broadcasting, and mobile TV. This course will also provide a systematic explanation of mobile computing as a discrete discipline and will provide an in-depth coverage of mobile systems and devices, mobile operating systems used for application development, mobile databases, client-server computing agents, application servers, security protocols, and mobile Internet, and ad-hoc and sensor networks.

Course Content

Unit I: 12 lecture hours

Introduction to Wireless Communication System: Evolution of mobile communications, Mobile Radio System around the world, Types of Wireless communication System, Comparison of Common wireless system, Trends in Cellular radio and personal communication. Second

generation Cellular Networks, Third Generation (3G) Wireless Networks , Wireless Local Loop(WLL), Wireless Local Area network(WLAN), Bluetooth and Personal Area Networks.

Unit II: 8 lecture hours

Cellular system, Hexagonal geometry cell and concept of frequency reuse, Channel Assignment Strategies Distance to frequency reuse ratio, Channel & co-channel interference reduction factor, S/I ratio consideration and calculation for Minimum Co-channel and adjacent interference, Handoff Strategies, Umbrella Cell Concept, Trunking and Grade of Service, Improving Coverage & Capacity in Cellular System-cell splitting, Cell sectorization , Repeaters, Micro cell zone concept, Channel antenna system design considerations.

Unit III: 12 lecture hours

Multiple Access Techniques: Introduction, Comparisons of multiple Access Strategies like TDMA,CDMA, FDMA, OFDM, and CSMA Protocols. Wireless Systems: GSM system architecture, Radio interface, Protocols, Localization and calling, Handover, Authentication and security in GSM, GSM speech coding, Concept of spread spectrum, Architecture of IS-95 CDMA system, Air interface, CDMA forward channels, CDMA reverse channels, Soft handoff, CDMA features, Power control in CDMA, Performance of CDMA System, RAKE Receiver, CDMA2000 cellular technology, GPRS system architecture.

Unit IV: 8 lecture hours

Recent trends: Introduction to Wi-Fi, WiMAX, ZigBee Networks, Software Defined Radio, UWB Radio, Wireless Adhoc Network and Mobile Portability, Security issues and challenges in a Wireless network.

Text Books

1. Wireless Communication, Theodore S. Rappaport, Prentice hall

Reference Books/Materials

1. Wireless Communications and Networking, Vijay Garg, Elsevie

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination

Examination Scheme:

Components	Quiz	Attendance	Mid Term Exam	Presentation/ Assignment/ etc.	End Term Exam

Weightage (%)	**10**	**10**	**20**	**10**	**50**

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Utilize mobile computing nomenclature to describe and analyze existing mobile computing frameworks and architectures.	**PO1, PO2**
CO2	Evaluate the effectiveness of different mobile computing frameworks.	**PO3, PO4**
CO3	Describe how mobile technology functions to enable other computing technologies.	**PO10, PSO1, PSO3**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3

ETCS410A	Mobile and Wireless Communication	2	2	2	2						3			3		3

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCS453A	**Mobile and Wireless Communication Lab**	L	T	P	C
Version 1.0		0	0	2	1
Pre-requisites/Exposure	Practical of computer communication				
Co-requisites	--				

Course Objectives

1. Use engineering knowledge to solve real world open-ended problems in wireless system design.
2. Use appropriate channel and traffic models to evaluate the impact of wireless service quality and capacity.
3. Generate solutions for complex design problems via proper choice of system parameters, analyze the results and make recommendations.
4. Design and develop software tools to perform the tasks required by the project; Identify the limitations and enhancements of the tools with respect to the project needs.

Course Outcomes

On completion of this course, the students will be able to

CO1. Learn to the syntax of Java Programming Language and implement applications in it.
CO2. Recognize features of object-oriented design such as encapsulation, polymorphism inheritance and composition of systems based on object identity.

CO3. Articulate re-usable programming components using Abstract Class, Interfaces and other permitted ways in packages.
CO4. Apply access control mechanism to safeguard the data and functions that can be applied by the object.
CO5. Understand multithreading and evaluate exception handing to create new applications.
CO6. Design GUI applications using pre-built frameworks available in Java.

Catalog Description

This course provides a comprehensive introduction to basic principles and techniques in cellular mobile communications. The topics include: communication overview and frequency reuse, the cellular concept, radio propagation environments, techniques of modulation and equalization, multiple access wireless systems: TDMA/FDMA systems, CDMA systems etc.

List of Experiments (Indicative)

1	To set up a satellite communication link & study of change in uplink & downlink frequency.	**2 lab hours**
2	To Study Transmission of Audio & Video Signals & Data communication over satellite link.	**2 lab hours**
3	To Study Transmission of telemetry data like temperature & light intensity over satellite link.	**2 lab hours**
4	To measure the propagation delay of signal in a Satellite communication link.	**2 lab hours**
5	To study different GPS data like longitude, latitude & different types of dilute of precision using GPS receiver.	**2 lab hours**
6	To study selection of various PN codes like Gold, Barker & MLS in CDMA technology.	**4 lab hours**
7	To study generation (spreading) & demodulation (Despreading) of DSSS modulated signal.	**4 lab hours**
8	To study Voice communication over DSSS.	**4 lab hours**
9	To study Minimum shift keying modulation & de modulation.	**4 lab hours**
10	FHSS Modulation & demodulation & transfer of numeric data.	**4 lab hours**

Modes of Evaluation: Quiz/Oral practical oral exam/presentation/projects/Practical Examination

Examination Scheme:

Components	Quiz	Attendance	Mid Term Exam	Presentation/ Assignment/ etc.	End Term Exam
Weightage (%)	10	10	20	10	50

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Use engineering knowledge to solve real world open-ended problems in wireless system design.	**PO1,PO2**
CO2	Use appropriate channel and traffic models to evaluate the impact of wireless service quality and capacity.	**PO3**
CO3	Generate solutions for complex design problems via proper choice of system parameters, analyze the results and make recommendations	**PO5**
CO4	Design and develop software tools to perform the tasks required by the project; identify the limitations and enhancements of the tools with respect to the project needs.	**PO8, PO9, PSO3**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex pr	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge

					obl em s											
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
ETCS453 A	Mobile and Wireless Communicati on Lab	2	3	3		2			2	3						3

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCS411A	**Machine Learning**	L	T	P	C
Version 1.0		3	0	0	3
Pre-requisites/Exposure	NIL				
Co-requisites					

Course Objectives

1. To develop an appreciation for what is involved in learning from data.
2. To understand a wide variety of learning algorithms.

3. To understand how to apply a variety of learning algorithms to data.
4. To understand how to perform evaluation of learning algorithms and model selection.
5. To become familiar with Dimensionality reduction Techniques.

Course Outcomes

On completion of this course, the students will be able to

CO1. Gain knowledge about basic concepts of Machine Learning
CO2. Identify machine learning techniques suitable for a given problem.
CO3. Solve the problems using various machine learning techniques.
CO4. Apply neural networks for suitable application.
CO5. Use a tool to implement typical clustering algorithms for different types of applications.
CO6. Apply Dimensionality reduction techniques.

Catalog Description

This course imparts comprehensive introduction to various topics in machine learning. It enables them to design and implement machine learning solutions to classification, regression, and clustering problems; and be able to evaluate and interpret the results of the algorithms.

Course Content

UNIT I **8 Hours**

Machine learning: overview and survey of its applications. Problem of induction and statistical inference: Input-output functions, Boolean functions, Parametric and nonparametric inference, Probability, uncertainty and Bayes theorem, Introduction to typical learning tasks: regression, pattern recognition, feature selection, classification, clustering, rule induction (association). Model validation techniques: cross-validation, leave-one-out, majority, Measures of performance (sensitivity, specificity, ROC curves, etc.)

UNIT II **8 Hours**

Dimensionality Reduction: Subset Selection, Shrinkage Methods, Principle Components Regression Linear Classification, Logistic Regression, Linear Discriminant Analysis Optimization, Classification-Separating Hyperplanes Classification

UNIT III **9 Hours**

Neural Networks: Non-linear Hypothesis, Biological Neurons, Model representation, Intuition for Neural Networks, Multiclass classification, Cost Function, Back Propagation Algorithm, Back Propagation Intuition, Weights initialization, Neural Network Training.

Support Vector Machines: Optimization Objective, Large Margin Classifiers, Kernels, SVM practical considerations

UNIT IV **10 Hours**

Supervised Learning: Additive model: logistic regression, Generative model: naïve Bayes classifier, Discriminative model: Decision trees, Neural networks.

Unsupervised Learning: Clustering: k-means, hierarchical, self-organizing map, EM algorithm, Feature selection principal component analysis.

Reinforcement Learning: Q-learning, Value function approximation, Policy search.

Text Books:

1. The Elements of Statistical Learning, T. Hastie, R. Tibshirani and J. H. Friedman, Springer.

Reference Books:

1. Joel Grus, "Data Science from Scratch: First Principles with Python", O'Reilly Media
2. Aurélien Géron, "Hands-On Machine Learning with Scikit-Learn and Tensor Flow: Concepts, Tools, and Techniques to Build Intelligent Systems", 1st Edition, O'Reilly Media
3. Jain V.K., “Data Sciences”, Khanna Publishing House, Delhi.
4. Jain V.K., “Big Data and Hadoop”, Khanna Publishing House, Delhi.
5. Jeeva Jose, “Machine Learning”, Khanna Publishing House, Delhi.
6. Chopra Rajiv, “Machine Learning”, Khanna Publishing House, Delhi.
7. Ian Goodfellow, Yoshua Bengio and Aaron Courville, "Deep Learning", MIT Press http://www.deeplearningbook.org
8. Jiawei Han and Jian Pei, "Data Mining Concepts and Techniques", Third Edition, Morgan Kaufmann Publisher

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination
Examination Scheme:

Components	Quiz	Attendance	Mid Term Exam	Presentation/ Assignment/ etc.	End Term Exam
Weightage (%)	10	10	20	10	50

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs

	Course Outcomes (COs)	Mapped Program Outcomes
CO1	Gain knowledge about basic concepts of Machine Learning	**PO1**
CO2	Identify machine learning techniques suitable for a given problem.	**PO4**
CO3	Solve the problems using various machine learning techniques.	**PO5**
CO4	Apply neural networks for suitable application.	**PO2**
CO5	Use a tool to implement typical clustering algorithms for different types of applications.	**PO3**
CO6	Apply Dimensionality reduction techniques.	**PO3**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2	PSO3
ETCS411A	Machine Learning	2	3	3	3	3								3		3

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCS455A	Machine Learning Lab	L	T	P	C
Version 1.0		0	0	2	1
Pre-requisites/Exposure	Programming for Problem Solving Lab				
Co-requisites	--				

Course Objectives

1. Develop the technical and practical skills to apply machine learning to solve real-world problems.
2. Explore regression as a supervised machine learning technique to predict a continuous variable (response or target) from a set of other variables (features or predictors)
3. Discover how variable selection and shrinkage methods are used to improve the efficiency of a regression model when applied to complex data sets
4. Explore classification as a supervised machine learning technique to predict binary (or discrete) response variables from a set of features
5. Understand what neural networks are, its most successful applications, and how it can be used within a business context

Course Outcomes

On completion of this course, the students will be able to

CO1. Understand the implementation procedures for the machine learning algorithms .

CO2. Design Java/Python programs for various Learning algorithms.

CO3. Apply appropriate data sets to the Machine Learning algorithms.

CO4. Identify and apply Machine Learning algorithms to solve real world problems.

Note: The programs can be implemented in either JAVA or Python.

1.For Problems 1 to 6 and 10, programs are to be developed without using the built-in classes or APIs of Java/Python.

2.Datasetscan be taken from standard repositories (https://archive.ics.uci.edu/ml/datasets.html) or constructed by the students.

Catalog Description

Machine Learning is concerned with computer programs that automatically improve their performance through experience. This course covers the theory and practical algorithms for

machine learning from a variety of perspectives. We cover topics such as FIND-S, Candidate Elimination Algorithm, Decision tree (ID3 Algorithm), Backpropagation Algorithm, Naïve Bayesian classifier, Bayesian Network, k-Means Algorithm, k-Nearest Neighbor Algorithm, Locally Weighted Regression Algorithm.

List of Experiments (Indicative)

1	Implement and demonstrate the FIND-S algorithm for finding the most specific hypothesis based on a given set of training data samples. Read the training data from a .CSV file.	**2 lab hours**
2	For a given set of training data examples stored in a .CSV file, implement and demonstrate the Candidate-Elimination algorithm to output a description of the set of all hypotheses consistent with the training examples.	**2 lab hours**
3	Write a program to demonstrate the working of the decision tree based ID3 algorithm. Use an appropriate data set for building the decision tree and apply this knowledge to classify a new sample.	**2 lab hours**
4	Build an Artificial Neural Network by implementing the Backpropagation algorithm and test the same using appropriate data sets.	**2 lab hours**
5	Write a program to implement the naïve Bayesian classifier for a sample training data set stored as a .CSV file. Compute the accuracy of the classifier, considering few test data sets.	**2 lab hours**
6	Assuming a set of documents that need to be classified, use the naïve Bayesian Classifier model to perform this task. Built-in Java classes/API can be used to write the program. Calculate the accuracy, precision, and recall for your data set.	**4 lab hours**
7	Write a program to construct a Bayesian network considering medical data. Use this model to demonstrate the diagnosis of heart patients using standard Heart Disease Data Set. You can use Java/Python ML library classes/API.	**4 lab hours**
8	Apply EM algorithm to cluster a set of data stored in a .CSV file. Use the same data set for clustering using k-Means algorithm. Compare the results of these two algorithms and comment on the quality of clustering. You can add Java/Python ML library classes/API in the program.	**4 lab hours**
9	Write a program to implement k-Nearest Neighbour algorithm to classify the iris data set. Print both correct and wrong predictions. Java/Python ML library classes can be used for this	**4 lab hours**

	problem.	
10	Implement the non-parametric Locally Weighted Regression algorithm in order to fit data points. Select appropriate data set for your experiment and draw graphs.	**4 lab hours**

Modes of Evaluation: Quiz/Oral practical oral exam/presentation/projects/Practical Examination

Examination Scheme:

Components	**Quiz**	**Attendance**	**Mid Term Exam**	**Presentation/ Assignment/ etc.**	**End Term Exam**
Weightage (%)	**10**	**10**	**20**	**10**	**50**

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Understand the implementation procedures for the machine learning algorithms.	**PO2**
CO2	Design Java/Python programs for various Learning algorithms.	**PO3**
CO3	Apply appropriate data sets to the Machine Learning algorithms.	**PO5**
CO4	Identify and apply Machine Learning algorithms to solve real world problems.	**PO8**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2	PSO3
ETCS455A	Machine Learning Lab		3	3		2			2							3

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCS517A	**Soft Computing**	L	T	P	C
Version 1.0		3		0	3
Pre-requisites/Exposure	Basics of fuzzy logic, neural network theory, and Genetic algorithms				
Co-requisites	--				

Course Objectives

The students will be able to get an idea on:

1. Neural Networks, architecture, functions and various algorithms involved.
2. Fuzzy Logic, Various fuzzy systems and their functions.

3. Genetic algorithms, its applications and advances.
4. The unified and exact mathematical basis as well as the general principles of various soft computing techniques.

Course Outcomes

Upon completion of the course the students will be able to:
CO1.Understand soft computing techniques and their role in problem solving.
CO2.Conceptualize and parameterize various problems to be solved through basic soft computing techniques.
CO3. Analyze and integrate various soft computing techniques in order to solve problems effectively and efficiently.
CO4.Develop application on different soft computing techniques like Fuzzy, GA and Neural network
CO5.Implement Neuro-Fuzzy and Neuro-Fuzz-GA expert system.
CO6. To understand the fundamental theory and concepts of neural networks, Identify different neural network architectures, algorithms, applications and their limitations.

Catalog Description

This course introduces soft computing methods which, unlike hard computing, are tolerant of imprecision, uncertainty and partial truth. The principal constituents of soft computing are fuzzy logic, neural network theory, and probabilistic reasoning.

Course Content

Unit I: **8 lecture hours**

Introduction: What is Soft Computing? Difference between Hard and Soft computing, Requirement of Soft computing, Major Areas of Soft Computing, Applications of Soft Computing. Neural Networks: What is Neural Network, Learning rules and various activation functions, Single layer Perceptrons, Back Propagation networks, Architecture of Backpropagation (BP) Networks, Backpropagation Learning, Variation of Standard Back propagation Neural Network, Introduction to Associative Memory, Adaptive Resonance theory and Self Organizing Map, Recent Applications.

Unit II: **12 lecture hours**

Fuzzy Systems: Fuzzy Set theory, Fuzzy versus Crisp set, Fuzzy Relation, Fuzzification, Minmax Composition, Defuzzification Method, Fuzzy Logic, Fuzzy Rule based systems, Predicate logic, Fuzzy Decision Making, Fuzzy Control Systems, Fuzzy Classification

Fuzzy Backpropagation Networks: LR type Fuzzy numbers, Fuzzy Neuron, Fuzzy BP Architecture, Learning in Fuzzy BP, Application of Fuzzy BP Networks.

Unit III: 12 lecture hours

Genetic Algorithm: History of Genetic Algorithms (GA), Working Principle, Various Encoding methods, Fitness function, GA Operators-Reproduction, Crossover, Mutation, Convergence of GA, Bit wise operation in GA, Multi-level Optimization.GA based Backpropagation Networks: GA based Weight Determination, K -factor determination in Columns.

Unit IV: 8 lecture hours

Hybrid Systems: Sequential Hybrid Systems, Auxiliary Hybrid Systems, Embedded Hybrid Systems, Neuro-Fuzzy Hybrid Systems, Neuro-Genetic Hybrid Systems, Fuzzy-Genetic Hybrid Systems.

Text Books

1. Neural Networks, Fuzzy Logic and Genetic Algorithms: Synthesis & Applications, S.Rajasekaran, G. A. Vijayalakshami, PHI.

Reference Books/Materials

1. Genetic Algorithms: Search and Optimization, E. Goldberg
2. Neuro-Fuzzy Systems, Chin Teng Lin, C. S. George Lee, PHI.
3. Build_Neural_Network_With_MS_Excel_sampleby Joe choong
4. S. N. Sivanandam& S.N. Deepa, "Principles of Soft Computing", Wiley, 2007
5. Rafik Aziz oglyAliev, R. R. Aliev: "Soft Computing and Its Applications", World Scientific, 2001

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination
Examination Scheme:

Components	Quiz	Attendance	Mid Term Exam	Presentation/ Assignment/ etc.	End Term Exam
Weightage (%)	10	10	20	10	50

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	Mapped Program Outcomes
CO1	Understand soft computing techniques and their role in problem solving.	PO1, PSO2

CO2	Conceptualize and parameterize various problems to be solved through basic soft computing techniques.	**PO3**
CO3	Analyze and integrate various soft computing techniques in order to solve problems effectively and efficiently.	**PO5**
CO4	Develop application on different soft computing techniques like Fuzzy, GA and Neural network	**PO2, PSO2, PSO3**
CO5	Implement Neuro-Fuzzy and Neuro-Fuzz-GA expert system.	**PO4**
CO6	To understand the fundamental theory and concepts of neural networks, Identify different neural network architectures, algorithms, applications and their limitations	**PO6**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2	PSO3
ETCS517 A	Soft Computing	2	2	2	3	3	2								3	3

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCS559A	Soft Computing Lab	L	T	P	C
Version 1.0		0	0	2	1
Pre-requisites/Exposure	Practical learning of Soft Computing				
Co-requisites	--				

Course Objectives

The students will be able to get an idea on:

1. Neural Networks, architecture, functions and various algorithms involved.
2. Fuzzy Logic, Various fuzzy systems and their functions.
3. Genetic algorithms, its applications and advances.
4. The unified and exact mathematical basis as well as the general principlesof various soft computing techniques.

Course Outcomes

Upon completion of the course the students will be able to:
CO1.Understand soft computing techniques and their role in problem solving.
CO2.Conceptualize and parameterize various problems to be solved through basic soft computing techniques.
CO3. Analyze and integrate various soft computing techniques in order to solve problems effectively and efficiently.
CO4.Develop application on different soft computing techniques like Fuzzy, GA and Neural network

Catalog Description

This course introduces soft computing methods which, unlike hard computing, are tolerant of imprecision, uncertainty and partial truth. The principal constituents of soft computing are fuzzy logic, neural network theory, and probabilistic reasoning.

Course Content

1	Create a perceptron with appropriate no. of inputs and outputs. Train using fixed increment learning algorithm until no change in weights is required. Output the final weights.	**2 lab hours**
2	Create a simple ADALINE network with appropriate no. of input and output nodes. Train using delta learning rule until no change in weights is required. Output the final weights.	**2 lab hours**
3	Train the autocorrelator by given patterns: A1= (-1,1, -1,1), A2=	**2 lab hours**

	(1,1,1, -1), A3=(-1, -1, -1, 1). Test it using patterns: Ax = (-1,1, -1,1), Ay= (1,1,1,1), Az= (-1, -1,-1,-1).	
4	Train the hetro-correlator using multiple training encoding strategy for given patterns: A1= (000111001) B1= (010000111), A2= (111001110) B2= (100000001), A3= (110110101) B3(101001010). Test it using pattern A2.	**2 lab hours**
5	Implement Union, Intersection, Complement and Difference operations on fuzzy sets. Also create fuzzy relation by Cartesian product of any two fuzzy sets and perform max-min composition on any two fuzzy relations.	**2 lab hours**
6	Solve Greg Viot's fuzzy cruise controller using MATLAB Fuzzy logic toolbox.	**4 lab hours**
7	Solve Air Conditioner Controller using MATLAB Fuzzy logic toolbox.	**4 lab hours**
8	Implement TSP using GA.	**4 lab hours**
9	Implement one applications for Adaptive Systems	**4 lab hours**
10	Implement fitness function, Cross over and mutation in GA algorithms.	**4 lab hours**
11	Implement genetic algorithm based backpropagation network in MATLAB.	

Modes of Evaluation: Quiz/Oral practical oral exam/presentation/projects/Practical Examination

Examination Scheme:

Components	**Quiz**	**Attendance**	**Mid Term Exam**	**Presentation/ Assignment/ etc.**	**End Term Exam**
Weightage (%)	**10**	**10**	**20**	**10**	**50**

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Understand soft computing techniques and their role in problem solving.	**PO2**

CO2	Conceptualize and parameterize various problems to be solved through basic soft computing techniques.	**PO3**
CO3	Analyze and integrate various soft computing techniques in order to solve problems effectively and efficiently.	**PO5, PO9**
CO4	Develop application on different soft computing techniques like Fuzzy, GA and Neural network	**PO4, PSO3**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
ETCS559 A	Soft Computing Lab		2	3	3	3				3						3

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCS 522A	Virtual Reality	L	T	P	C
Version 1.0		3	0	0	3
Pre-requisites/Exposure	Basics of Computer Graphics				
Co-requisites	Data Structure and Programming				

Course Objectives

The Objective of this course is to provided tailed understanding of the concepts of Virtual Reality and its applications.

Course Outcomes

At the end of the course ,the students will be able to:
CO1. Understand geometric modelling and Virtual environment.
CO2. Study about Virtual Hardware and Software
CO3. Develop Virtual Reality applications

Catalog Description

This course is designed to introduce students to the field of virtual reality (VR) and provide students with hands-on experience developing applications for modern virtual and augmented reality systems. In the course, students learn about the historical development of virtual reality technology and virtual reality as a research field, gain mastery of fundamental principles, algorithms, and design patterns in computer graphics, discover the perceptual science behind mixed reality technologies, and explore libraries and tools for creating VR experiences

Course Content

Unit I: (6 hours)

Introduction to Virtual Reality, Virtual Reality and Virtual Environment: Introduction, Computer graphics, Real time computergraphics,FlightSimulation,Virtualenvironmentrequirement,benefitsofvirtualreality,Historicaldevelopment of VR, Scientific Landmark

3D Computer Graphics: Introduction, The Virtual world space, positioning the virtual observer, the perspective projection, human vision, stereo perspective projection, 3D clipping, Color theory, Simple 3D modelling, Illumination models, Reflection models, Shading algorithms, Radiosity, Hidden Surface Removal, Realism-Stereographic image.

Unit II: **12 lecture hours**

GeometricModelling:Introduction,From2Dto3D,3Dspacecurves,3D boundary representation

Geometrical Transformations: Introduction, Frames of reference, Modelling transformations, Instances, Picking, Flying, Scaling the VE, Collision detection

Generic VR system: Introduction, Virtual environment, Computer environment, VR technology, Model of interaction, VR Systems.

Unit III: **10 lecture hours**

Animating the Virtual Environment: Introduction, The dynamics of numbers, Linear and Non-linear interpolation, the animation of objects, linear and non-linear translation, shape & object in betweening, free from deformation, particle system.
PhysicalSimulation:Introduction,Objectsfallinginagravitationalfield,Rotatingwheels,Elasticcollisions,projectiles, simple pendulum, springs, Flight dynamics of an aircraft.

Unit IV: **10 lecture hours**

Human factors: Introduction, the eye ,the ear ,the somatic senses.

VRHardware:Introduction,sensorhardware,Head-coupleddisplays,Acoustichardware,Integrated VR systems.
VRSoftware:Introduction,Modellingvirtualworld,Physicalsimulation,VRtoolkits,IntroductiontoVRML

Unit V: **7 lecture hours**

VR Applications: Introduction,Engineering,Entertainment,Science,Training.TheFuture: Virtual environment ,modes of interaction

LISTOFSUGGESTEDTEXTBOOKS

1. JohnVince,"VirtualRealitySystems",PearsonEducationAsia,2007.
2. AnandR.," Augmented and Virtual Reality", Khanna Publishing House, Delhi.
3. Adams,"VisualizationsofVirtualReality",TataMcGrawHill,2000.
4. GrigoreC. Burdea, Philippe Coiffet, "Virtual Reality Technology", Wiley Inter Science, 2nd Edition, 2006.
5. WilliamR.Sherman,AlanB.Craig,"UnderstandingVirtualReality:Interface,Applicationand Design", Morgan Kaufmann,2008.

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination

Examination Scheme:

Components	Quiz	Attendance	Mid Term Exam	Presentation/ Assignment/ etc.	End Term Exam
Weightage (%)	10	10	20	10	50

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Understand geometric modelling and Virtual environment.	**PO1, PO2**
CO2	Study about Virtual Hardware and Software	**PO5, PO7**
CO3	Develop Virtual Reality applications.	**PO3, PO12**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3

ETCS522 A	Virtual Reality	2	3	3		3		2						3		3

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCS564A	**Virtual Reality Lab**	L	T	P	C
Version 1.0		0	0	2	1
Pre-requisites/Exposure	**Pre-requisites/Exposure**				
Co-requisites	**Co-requisites**				

Course Objectives

The objective of this course is to provide detailed understanding of the concepts of Virtual Reality and its applications.

Course Outcomes

At the end of the course, the students will be able to:
CO1. Understand geometric modelling and Virtual environment.
CO2. Study about Virtual Hardware and Software
CO3. Develop Virtual Reality applications

Catalog Description

This course complements ETCS 564A. The course presents students with individual assignments and a semester long team project to develop a fully- functional applications.

List of Experiments (Indicative)

1	Developing architecture of a house using Virtual Reality	**2 lab hours**

2	Perform CRO based experiment using Virtual Reality.	**2 lab hours**
3	Undertaking qualitative analysis in Chemistry using Virtual Reality	**2 lab hours**
4	Carry out assembly/disassembly of an engine using Virtual Reality.	**2 lab hours**
5	Explore human an atomy using Virtual Reality.	**2 lab hours**
6	Simulation of circulation of blood in heart.	**4 lab hours**
7	Simulation of Fight/Vehicle/Space Station.	**4 lab hours**
8	Building Electronic circuit using Virtual Reality, given basic electronic components.	**4 lab hours**
9	Developing concept of Virtual classroom with multiplayer.	**4 lab hours**

Modes of Evaluation: Quiz/Oral practical oral exam/presentation/projects/Practical Examination

Examination Scheme:

Components	**Quiz**	**Attendance**	**Mid Term Exam**	**Presentation/ Projects/ etc.**	**End Term Exam**
Weightage (%)	**10**	**10**	**20**	**10**	**50**

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Understand geometric modelling and Virtual environment.	**PO1, PO2**
CO2	Study about Virtual Hardware and Software	**PO5, PO7**
CO3	Develop Virtual Reality applications	**PO3, PO12**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2	PSO3
ETCS564A	Virtual Reality Lab	2	3	3		3		2						3		3

1= weakly mapped
2= moderately mapped
3=strongly mapped

ETCS515A	**Etical Hacking**	L	T	P	C
Version 1.0		3	0	0	3
Pre-requisites/Exposure	Basics Algebra				
Co-requisites	--				

Course Objectives

1. To learn penetration testing.
2. To learn difference between threat, vulnerability and attacks.
3. To learn security mechanisms.
4. To learn different types of attacks.
5. To implement tools and methods to improve the security of the system from hackers.
6. To differentiate between authorised and unauthorised users.
7. To understand latest mechanism of secure and safe network.

Course Outcomes

On completion of this course, the students will be able to

CO1. To learn ethical considerations of hacking

CO2. To learn legal considerations of hacking

CO3. To collect information using network scanning

CO4. Identify methods to gain access to systems

CO5. Analyze social engineering methods

CO6. Explain common physical security weaknesses

Catalog Description

This class will immerse the students into an interactive environment where they will be shown how to scan, test, hack and secure their own systems. The lab intensive environment gives each student in-depth knowledge and practical experience with the current essential security systems. Students will begin by understanding how perimeter defenses work and then be lead into scanning and attacking their own networks, no real network is harmed.

Course Content

Unit I: 12 lecture hours

Introduction to Ethical Hacking: Five phases of ethical hacking, different types of hacker attacks, Foot printing and Reconnaissance, Scanning Networks, TCP flag types, types of port scans, scanning countermeasures

Unit II: 8 lecture hours

Enumeration: Role and enumeration techniques recognize how to establish a sessions, Identify

enumeration countermeasures, Perform active and passive enumeration. Sniffers, types of sniffing and protocols vulnerable to sniffing, Recognize types of sniffing attacks, methods for detecting sniffing, different types of social engineering, and social engineering countermeasures.

Unit III: **12 lecture hours**

System Hacking: Identify different types of password attacks, Use a password cracking tool, Identify various password cracking countermeasures, Identify different ways to hide files, Recognize how to detect a rootkit, Identify tools that can be used to cover attacker tracks.

Unit IV: **8 lecture hours**

Trojans and Backdoors: Concept of Trojan infects a system, ports used by Trojans and Trojan countermeasures, symptoms of a virus and its working, Detection methods and virus countermeasures.

Text Books: 1. A Beginners Guide To Hacking Computer Systems

Reference Books:

1. Black Book of Viruses and Hacking 2. Secrets of Super and Professional Hackers 3. Dangerous Google Hacking Database and Attacks 4. Internet Advanced Denial of Service (DDOS) Attack 5. Computer Hacking & Malware Attacks for Dummies

Components	Quiz	Attendance	Mid Term Exam	Presentation/ Assignment/ etc.	End Term Exam
Weightage (%)	10	10	20	10	50

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	Mapped Program Outcomes
CO1	To learn ethical consideration of hacking	PO2
CO2	To learn legal consideration of hacking	PO3
CO3	To collect information using network scanning	PO4

CO4	Identify methods to gain access to systems	**PO5**
CO5	Analyze social engineering methods	**PO4**
CO6	Explain common physical security weaknesses	**PO4, PO9, PSO1, PSO3**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
ETCS515A	Ethical hacking		2	3	3	3				3				3		2

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCS557A	Ethical Hacking Lab	L	T	P	C
Version 1.0		0	0	2	1
Pre-requisites/Exposure	Basic algebra				
Co-requisites	--				

Course Objectives

1. To understand the various security issues.
2. To learn different tools and techniques in ethical hacking.
3. To implement security tools.

Course Outcomes

On completion of this course, the students will be able to

CO1. Identify and analyse the stages an ethical hacker requires to take in order to compromise a target system.

CO2. Identify tools and techniques to carry out a penetration testing.

CO3. Critically evaluate security techniques used to protect system and user data.

CO4. Demonstrate systematic understanding of the concepts of security at the level of policy and strategy in a computer system.

Catalog Description

This course is hands-on application of security tools to test network and systems security. The course focuses on hacking techniques and technology from an offensive perspective. The student will learn to scan, test, hack and secure systems. Students will learn the five phases of ethical hacking: reconnaissance; gaining access; enumeration; maintaining access; and covering their tracks. Throughout the course, students will be immersed in a hacker's mindset, evaluating not just logical, but physical security exploring every possible point of entry to find the weakest link in an organization.

Course Content

1	Implementation on various phases of Ethical hacking.	**2 lab hours**

2	Implementation on networking concept.	**2 lab hours**
3	Implementation on Foot Printing	**2 lab hours**
4	Case Study on Windows linux system security.	**2 lab hours**
5	Implementation on Proxy server	**2 lab hours**
6	Implementation on System hacking and security.	**2 lab hours**
7	Implementation on Windows Linux scripting.	**2 lab hours**
8	Implementation on Network hacking and security.	**2 lab hours**
9	Implementation on Foot Printing and Information gathering.	**2 lab hours**
10	Case study on Google hacking.	**2 lab hours**
11	Case study on Hacking attacks	**2 lab hours**
12	Case study on Web application hacking.	**2 lab hours**
13	Case study on Cryptography	**2 lab hours**
14	Case study on Honeypots	**2 lab hours**
15	Implementation on Wireless and mobile hacking and security.	**2 lab hours**

Modes of Evaluation: Quiz/Oral practical oral exam/presentation/projects/Practical Examination

Examination Scheme:

Components	**Quiz**	**Attendance**	**Mid Term Exam**	**Presentation/ Assignment/ etc.**	**End Term Exam**

Weightage (%)	**10**	**10**	**20**	**10**	**50**											

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Identify and analyze the stages an ethical hacker requires to take in order to compromise a target system.	**PO2**
CO2	Identify tools and techniques to carry out a penetration testing.	**PO3**
CO3	Critically evaluate security techniques used to protect system and user data.	**PO5, PSO1, PSO3**
CO4	Demonstrate systematic understanding of the concepts of security at the level of policy and strategy in a computer system.	**PO9**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2	PSO3

ETCS557 A	Ethical Hacking Lab		2	3		3				3				3		2

1=weakly mapped
2= moderately mapped
3=strongly mapped

Semester VI

ETCA324A	.Net Framework	L	T	P	C
Version 1.0		3	1	0	4
Pre-requisites/Exposure	Knowledge of C and C++				
Co-requisites	--				

Course Objectives

1. To provide a consistent object-oriented programming environment whether object code is stored and executed locally, executed locally but Internet-distributed, or executed remotely.
2. To provide a code-execution environment that minimizes software deployment and versioning conflicts.
3. To provide a code-execution environment that guarantees safe execution of code, including code created by an unknown or semi-trusted third party.
4. To provide a code-execution environment that eliminates the performance problems of scripted or interpreted environments.
5. To make the developer experience consistent across widely varying types of applications, such as Windows-based applications and Web-based applications.

Course Outcomes

On completion of this course, the students will be able to

CO1. Introduction to the .NET framework.
CO2. Students will describe the basic structure of a Visual Basic.NET project and use main features of the integrated development environment (IDE)
CO3. Learn about ASP. NET controls and create applications using Microsoft Windows Forms
CO4. Students will create applications that use ADO. NET
CO5. To understand and be able to explain Security in the .NET framework and Deployment in the .NET.

Catalog Description

In this course we will learn the fundamentals of the .Net framework, gaining a deeper understanding of web application standards, tools and techniques.

Course Content

Unit I: **10 lecture hours**

Introduction to .NET technologies: Features of .NET, .NET Framework, CLR, MSIL, .NET class library, .NET Languages, CTS, assemblies, manifest, and metadata, what is ASP.NET? Difference between ASP and ASP.NET.

Unit II: **10 lecture hours**

Controls in ASP.NET: Overview of Dynamic Web page, Understanding ASP.NET Controls, Applications, Web servers, Installation of IIS. Web forms, web form controls -server controls, client controls. Adding controls to a web form, Buttons, Text Box, Labels, Checkbox, Radio Buttons, List Box. Adding controls at runtime. Running a web Application, creating a multiform web project. Form Validation: Client-side validation, server-Side validation, validation Controls: Required Field Comparison Range. Calendar control, Ad rotator Control, Internet Explorer Control.

Unit III: **10 lecture hours**

Overview of ADO.NET and XML: What is ADO.NET, from ADO to ADO. NET. ADO.NET architecture, Accessing Data using Data Adapters and Datasets, using Command & Data Reader, binding data to data bind Controls, displaying data in data grid, XML basics, attributes, fundamental XML classes: Document, text writer, text reader. XML validations, XML in ADO.NET, XML Data Document.

Unit IV: **10 lecture hours**

ASP.NET Applications: Creating, tracking, caching, error handling, Securing ASP.NET applications - form based applications, window-based application, State management- View state, Session state, Application state, Building ASP.NET web services, working with ASP.NET applications, creating custom controls.

Text Books

1. Stephen Walther, “ASP.NET Unleashed”, SAMS publications

Reference Books/Materials

1.ASP.NET, WroxPublications
2.ASP.NET and VB.NET, Wrox Publication

3.ASP.NET and C#.NET, Wrox Publication.

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination
Examination Scheme:

Components	Quiz	Attendance	Mid Term Exam	Presentation/ Assignment/ etc.	End Term Exam
Weightage (%)	10	10	20	10	50

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Introduction to the .NET framework.	**PO1, PO2**
CO2	Students will describe the basic structure of a Visual Basic/C#.NET project and use main features of the integrated development environment (IDE)	**PO3**
CO3	Learn about ASP. NET controls and create applications using Microsoft Windows Forms	**PO5**
CO4	Students will create applications that use ADO. NET	**PO5**
CO5	To understand and be able to explain Security in the .NET framework and Deployment in the .NET	**PO4**

		Engineering Knowledge	Problem analysis	Design/ development of solutions	Conduct investigations of complex problems	Modern tool usag	The engineer an	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behavior	Knowledge

						e	d society									
Course Code	Course Title	PO 1	PO 2	PO 3	PO4	PO 5	PO 6	PO7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2	PSO3
ETCA324A	.Net Framework	2	2	3	2	3								3		2

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETC520A	**Internet Technologies**	L	T	P	C
Version 1.0		3	1	0	4
Pre-requisites/Exposure	Basics of Internet				
Co-requisites	--				

Course Objectives

1. To understand the terms related to the Internet and how the Internet is changing the world.
2. To understand how computers are connected to the Internet and demonstrate the ability to use the World Wide Web.
3. 3. Demonstrate an understanding of and the ability to use electronic mail and other internet based services
4. 4. Understand the design principles of Web pages and how they are created
5. 5. To develop an ability to create basic Web pages with HTML.

Course Outcomes

On completion of this course, the students will be able to

CO1. Analyze a web page and identify its elements and attributes.
CO2.Create web pages using XHTML and Cascading Style Sheets.
CO3.Build dynamic web pages using JavaScript (Client side programming).
CO4. Create XML documents and Schemas.
CO5. Build interactive web applications using AJAX.
CO6. Expose students to the basic tools and applications used in Web publishing.
CO7. Provide internet connection to the system and its installation.
CO8. Suggest appropriate routing algorithm for the network.

Catalog Description

Course Content

Unit I: **12 lecture hours**

Introduction: Overview, Network of Networks, Intranet, Extranet and Internet. World Wide Web, Domain and Sub domain, Address Resolution, DNS, Telnet, FTP, HTTP. Review of TCP/IP: Features, Segment, Three-Way Handshaking, Flow Control, Error Control, Congestion control. IP Datagram, IPv4 and IPv6. IP Subnetting and addressing: Classful and Classless Addressing, Subnetting. NAT, IP masquerading, IP tables. Internet Routing Protocol: Routing -Intra and Inter Domain Routing, Unicast and Multicast Routing, Broadcast. Electronic Mail: POP3, SMTP.

Unit II: **8 lecture hours**

PERL: Introduction, Variable, Condition, Loop, Array, Implementing data structure, Hash, String, Regular Expression, File handling, I/O handling. JavaScript: Basics, Statements, comments, variable, comparison, condition, switch, loop, break. Object - string, array, Boolean, reg-ex. Function, Errors, Validation. Cookies: Definition of cookies, Create and Store a cookie with example. Java Applets: Container Class, Components, Applet Life Cycle, Update method; Parameter passing applet, Applications.

Unit III: **12 lecture hours**

Client-Server programming In Java: Java Socket, Java RMI. Threats: Malicious code-viruses, Trojan horses, worms; eavesdropping, spoofing, modification, denial of service attacks. Network security techniques: Password and Authentication; VPN, IP Security, security in electronic transaction, Secure Socket Layer (SSL), Secure Shell (SSH). Firewall: Introduction, Packet filtering, Stateful, Application layer, Proxy.

Unit IV: **8 lecture hours**

Internet Telephony: Introduction, VoIP. Multimedia Applications: Multimedia over IP: RSVP, RTP, RTCP and RTSP. Streaming media, Codec and Plugins, IPTV. mywbut.com Search Engine and Web Crawler: Definition, Meta data, Web Crawler, Indexing, Page rank, overview of SEO.

Text Books

1. Web Technology: A Developer's Perspective, N.P. Gopalan and J. Akilandeswari, PHI, Learning, Delhi, 2013.

Reference Books/Materials

1. Internetworking Technologies, An Engineering Perspective, Rahul Banerjee, PHI Learning, Delhi, 2011.

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination
Examination Scheme:

Components	Quiz	Attendance	Mid Term Exam	Presentation/ Assignment/ etc.	End Term Exam
Weightage (%)	10	10	20	10	50

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	Mapped Program Outcomes
CO1	Analyze a web page and identify its elements and attributes.	PO2
CO2	Create web pages using XHTML and Cascading Style Sheets.	PO3

CO3	Build dynamic web pages using JavaScript (Client side programming).	**PO4**
CO4	Create XML documents and Schemas.	**PO5**
CO5	Build interactive web applications using AJAX.	**PO4**
CO6	Expose students to the basic tools and applications used in Web publishing.	**PO4**
CO7	Provide internet connection to the system and its installation.	**PO9**
CO8	Suggest appropriate routing algorithm for the network.	**PSO3**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
ETC520 A	INTERNET TECHNOLOGIES		2	3	3	3				3						3

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCS401A	Artifical Intelligence	L	T	P	C
Version 1.0		3	1	0	4
Pre-requisites/Exposure	Basics of Computer Programming				
Co-requisites	--				

Course Objectives

1. To have clear understanding of the problem-solving processes.
2. To explore Search strategies ranging from blind or uninformed search to heuristic or informed search are discussed.
3. To understand real world always entails uncertainty and the concept of uncertainty is introduced.
4. To know about Probabilistic reasoning, representing knowledge under uncertainty, Bayesian Networks, Exact and approximate inference in Bayesian Networks
5. To gain idea of supervised, unsupervised and reinforcement learning is covered.
6. To introduce the students to the challenges involved in designing intelligent

Course Outcomes

On completion of this course, the students will be able to

CO1. Understand the various searching techniques, constraint satisfaction problem and example problems- game playing techniques.
CO2. Apply these techniques in applications which involve perception, reasoning and learning.
CO3. Explain the role of agents and how it is related to environment and the way of evaluating it and how agents can act by establishing goals.
CO4. Acquire the knowledge of real world Knowledge representation.
CO5. Analyze and design a real world problem for implementation and understand the dynamic behavior of a system.
CO6. Use different machine learning techniques to design AI machine and enveloping applications for real world problems.
CO7.Demonstrate an ability to share in discussions of AI, its current scope and limitations, and societal implications.

Catalog Description

The course introduces the theoretical building blocks necessary to create intelligent machines. While we may struggle to define intelligence in an absolute sense, we can agree upon multiple approaches toward creating AI; from an initial attempt at acting humanly to a broader context of

acting rationally. Solving problems which are seemingly simple for humans can seem like insurmountable hurdles for machines.

Course Content

Unit I: **8 lecture hours**

Scope of AI: Games, theorem proving, natural language processing, vision and speech processing, robotics, expert systems, AI techniques-search knowledge, abstraction. Problem Solving (Blind): State space search; production systems, search space control; depthfirst, breadth-first search. Heuristic Based Search: Heuristic search, Hill climbing, best-first search, A* Algorithm, Problem Reduction, Constraint Satisfaction

Unit II: **12 lecture hours**

Knowledge Representation: Predicate Logic: Unification, Modus Ponens, Modus Tokens, Resolution in Predicate Logic, Conflict Resolution Forward Chaining, Backward Chaining, Declarative and Procedural Representation, Rule based Systems. Structured Knowledge Representation: Semantic Nets: Slots, exceptions and default frames, conceptual dependency

Unit III: **12 lecture hours**

Handling Uncertainty: Non-Monotonic Reasoning, Probabilistic reasoning: Bayesian Inference, use of uncertainty factors. Natural Language Processing: Introduction, Syntactic Processing, Semantic Processing, Pragmatic Processing.

Unit IV: **8 lecture hours**

Learning: Concept of learning, learning automation, genetic algorithm, learning by inductions, neural nets. Expert Systems: Need and justification for expert systems, knowledge acquisition, Case Studies: MYCIN, RI.

Text Books

1. Artificial Intelligence, E. Rich and K. Knight, TMH.

Reference Books/Materials

1. Artificial Intelligence, P. H. Winston,Pearson Education.
2. Introduction to AI and Expert Systems, D. W. Patterson, PHI.

3. Principles of AI, N. J. Nilsson, Narosa Publishing House

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination
Examination Scheme:

Components	Quiz	Attendance	Mid Term Exam	Presentation/ Assignment/ etc.	End Term Exam
Weightage (%)	10	10	20	10	50

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Understand the various searching techniques, constraint satisfaction problem and example problems- game playing techniques.	**PO1**
CO2	Apply these techniques in applications which involve perception, reasoning and learning.	**PO4**
CO3	Explain the role of agents and how it is related to environment and the way of evaluating it and how agents can act by establishing goals.	**PO5**
CO4	Acquire the knowledge of real world Knowledge representation.	**PO2**
CO5	Analyze and design a real world problem for implementation and understand the dynamic behavior of a system.	**PO3**
CO6	Use different machine learning techniques to design AI machine and enveloping applications for real world problems.	**PO3**
CO7	Demonstrate an ability to share in discussions of AI, its current scope and limitations, and societal implications.	**PSO1**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2	PSO3
ETCS401A	ARTIFICAL INTELLIGENCE	2	3	2	3	3								3		

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCS451A	**Artificial Intelligence Lab**	L	T	P	C
Version 1.0		0	0	2	1
Pre-requisites/Exposure	Basics of Prolog/ Python				
Co-requisites	--				

Course Objectives

1. To have clear understanding of the problem-solving processes.

2. To explore Search strategies ranging from blind or uninformed search to heuristic or informed search are discussed.
3. To understand real world always entails uncertainty and the concept of uncertainty is introduced.
4. To know about Probabilistic reasoning, representing knowledge under uncertainty, Bayesian Networks, Exact and approximate inference in Bayesian Networks
5. To gain idea of supervised, unsupervised and reinforcement learning is covered.
6. To introduce the students to the challenges involved in designing intelligent

Course Outcomes

On completion of this course, the students will be able to
CO1. Demonstrate working knowledge in Prolog in order to write simple Prolog programs
CO2. Understand different types of AI agents know various AI search algorithms (uninformed, informed, heuristic, constraint satisfaction, genetic algorithms)
CO3. Understand the fundamentals of knowledge representation (logic-based, frame-based, semantic nets), inference and theorem proving
CO4. Know how to build simple knowledge-based systems
CO5.Demonstrate working knowledge of reasoning in the presence of incomplete and/or uncertain information

Catalog Description

While AI applications can be developed in any number of different languages, certain language features make programming AI applications straightforward. Prolog is structured in such a way that AI program development is supported by Prolog language features. Other languages, such as Java, support AI programming through code libraries. This course will provide students with an introduction to AI via programming features that support basic AI applications. The main of this course is make students familiar with AI programming and be able to use it in future models to implement various AI applications.

List of Experiments (Indicative)

1	Write a program to solve 8-queens problem in Prolog.	**2 lab hours**
2	Solve any problem using depth first search in Prolog.	**2 lab hours**

3	Solve any problem using best first search in Prolog.	**2 lab hours**
4	Solve 8-puzzle problem using best first search in Prolog.	**2 lab hours**
5	Solve Robot (traversal) problem using means End Analysis.	**2 lab hours**
6	Solve traveling salesman problem in Prolog.	**2 lab hours**
7	Write a Program to Implement Tic-Tac-Toe game in Prolog/python.	**2 lab hours**
8	Write a Program to Implement Water-Jug problem.	**3 lab hours**
9	Write a Program to Implement Monkey Banana Problem using Python.	**2 lab hours**
10	Write a Program to Implement N-Queens Problem.	**4 lab hours**
11	Write a Program to Implement Missionaries-Cannibals Problems.	**4 lab hours**
14	Make a minor project using AI.	**3 lab hours**
15	Study about various applications of AI.	**4 lab hours**

Modes of Evaluation: Quiz/Oral practical oral exam/presentation/projects/Practical Examination

Examination Scheme:

Components	**Quiz**	**Attendance**	**Mid Term**	**Presentation/**	**End Term**

			Exam	Assignment/ etc.	Exam
Weightage (%)	**10**	**10**	**20**	**10**	**50**

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Demonstrate working knowledge in Prolog in order to write simple Prolog programs	**PO1**
CO2	Understand different types of AI agents know various AI search algorithms (uninformed, informed, heuristic, constraint satisfaction, genetic algorithms)	**PO4**
CO3	Understand the fundamentals of knowledge representation (logic-based, frame-based, semantic nets), inference and theorem proving	**PO5**
CO4	Know how to build simple knowledge-based systems	**PO2**
CO5	Demonstrate working knowledge of reasoning in the presence of incomplete and/or uncertain information.	**PSO3**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS	PS	PS

														O1	O2	O3
ETCS451A	ARTIFICIAL INTELLIGENCE LAB	2	3		3	3										3

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCA364A	**Net Framework Lab**	L	T	P	C
Version 1.0		0	0	2	1
Pre-requisites/Exposure	Knowledge of C and C++				
Co-requisites	--				

Course Objectives

1. To learn the basics of .net Frame work and C# language
2. To learn C# elements and OOPS concepts
3. To learn interface and inheritance concepts in C# language
4. To learn fundamentals of window application programming and create a window application
5. To develop web applications and learn advanced features of C#

Course Outcomes

On completion of this course, the students will be able to:

CO1. Introduction to the .NET framework.
CO2. Students will describe the basic structure of a Visual Basic.NET project and use main features of the integrated development environment (IDE)
CO3. Learn about ASP. NET controls and create applications using Microsoft Windows Forms
CO4. Students will create applications that use ADO. NET

CO5. To understand and be able to explain Security in the .NET framework and Deployment in the .NET.

Catalog Description

Based on theory subject **ETCS 324A**, the following experiments are to be performed. It enables students to understand the concept of .net Framework and create applications.

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination
Examination Scheme:

Components	Quiz	Attendance	Mid Term Exam	Presentation/ Assignment/ etc.	End Term Exam
Weightage (%)	10	10	20	10	50

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Introduction to the .NET framework.	**PO1, PO2**
CO2	Students will describe the basic structure of a Visual Basic.NET project and use main features of the integrated development environment (IDE)	**PO3**
CO3	Learn about ASP. NET controls and create applications using Microsoft Windows Forms	**PO5**
CO4	Students will create applications that use ADO. NET	**PO5**
CO5	To understand and be able to explain Security in the .NET framework and Deployment in the .NET.	**PO4**

List of Experiments (Indicative)

1	1. Write a program using web controls to a)Factorial of a number b) Money Conversion c) Quadratic Equation d) Temperature Conversion e) Login Control	**4 lab hours**
2	Write a program for Ad rotator Control	**4 lab hours**
3	2. Write a program for Calendar control a) Display a message in calendar b) Display vacations in calendar c) Select a day in calendar control using style	**4 lab hours**
4	Write a program for Tree view control and use various operation of Tree view control	**4 lab hours**
5	Write a program to design graphical user interface and display records stored in database	**4 lab hours**
6	Write a program to insert and delete the records in database	**4 lab hours**
7	Write a program of Data binding using drop down list control	**4 lab hours**
8	Design a interactive website for admissions in university.	**4 lab hours**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behavior	Knowledge

							society									
Course Code	Course Title	PO 1	PO 2	PO 3	PO4	PO 5	PO 6	PO7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2	PSO3
ETCA364A	.Net Framework lab	2	2	3	2	3						3		3		2

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCA519A	**Blockchain**	L	T	P	C
Version 1.0		3	1	0	4
Pre-requisites/Exposure	Basics of Cryptography				
Co-requisites	Basic Mathematics				

Course Objectives

1. Help in understanding Creation of block and working of blockchain technology to innovate and improve business process.

Course Outcomes

On completion of this course, the students will be able to

CO1. Understand blockchain technology.
CO2. Develop blockchain based solutions and write smart contract using Hyper ledger Fabric and Ethereum frameworks.

CO3. Build and deploy block chain application for on premise and cloud-based architecture.
CO4. Integrate ideas from various domains and implement them using block chain technology in different perspectives.

Catalog Description

Through this subject, student will be able to understand the coarse grained aspects of Blockchain Technology. Student will understand the applications of Blockchain and its working in networks. The internals of framework and working will be discussed throughout the course duration.

Course Content

Unit I: **8 lecture hours**

Introduction: Overview of Block chain, Public Ledgers, Bitcoin, Smart Contracts, Block in a Blockchain, Transactions, Distributed Consensus, Public vs Private Block chain, Understanding Cryptocurrency to Block chain, Permissioned Model of Block chain, Overview of Security aspects of Blockchain.

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

Unit II: **12 lecture hours**

Bitcoin and Blockchain: Creation of coins, Payments and double spending, Bitcoin Scripts, Bitcoin P2P Network, Transaction in Bitcoin Network, Block Mining, Block propagation and block relay.

Working with Consensus in Bitcoin: Distributed consensus in open environments, Consensus in a Bitcoin network, Proof of Work (PoW) – basic introduction, Hashcash PoW, Bitcoin PoW, Attacks on PoW and the monopoly problem, Proof of Stake, Proof of Burn and Proof of Elapsed Time, The life of a Bitcoin Miner, Mining Difficulty, Mining Pool.

Unit III: **12 lecture hours**

Permissioned Blockchain: Permissioned model and use cases, Design issues for Permissioned block chains, Execute contracts, State machine replication, Overview of Consensus models for permissioned block chain- Distributed consensus in closed environment, Paxos, RAFT Consensus, Byzantine general problem, Byzantine fault tolerant system, Lamport-Shostak-Pease BFT Algorithm, BFT over Asynchronous systems.

Enterprise application of Blockchain: Cross border payments, Know Your Customer (KYC), Food Security, Mortgage over Block chain, Block chain enabled Trade, We Trade – Trade Finance Network, Supply Chain Financing, Identity on Block chain

Unit IV: **10 lecture hours**

Hyperledger Fabric: Architecture, Identities and Policies, Membership and Access Control, Channels, Transaction Validation, Writing smart contract using Hyperledger Fabric, Writing smart contract using Ethereum, Overview of Ripple and Corda

Text Books

1. Melanie Swan, "Block Chain: Blueprint for a New Economy", O'Reilly, 2015
2. Josh Thompsons, "Block Chain: The Block Chain for Beginners- Guide to Blockchain Technology and Leveraging Block Chain Programming"
3. Daniel Drescher, "BlockChain Basics", Apress; 1st edition, 2017
4. AnshulKaushik, "Block Chain and Crypto Currencies", Khanna Publishing House, Delhi.
5. Imran Bashir, "Mastering Block Chain: Distributed Ledger Technology, Decentralizationand Smart Contracts Explained", Packt Publishing
6. RiteshModi, "Solidity Programming Essentials: A Beginner's Guide to Build SmartContracts for Ethereum and Block Chain", Packt Publishing

Reference Books/Materials

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination
Examination Scheme:

Components	Quiz	Attendance	Mid Term Exam	Presentation/ Assignment/ etc.	End Term Exam
Weightage (%)	10	10	20	10	50

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	Mapped Program Outcomes
CO1	Understand block chain technology.	PO1

CO2	Develop blockchain based solutions and write smart contract using Hyperledger Fabricand Ethereum frameworks	**PO2, PO3**
CO3	Build and deploy block chain application for on premise and cloud-based architecture	**PO5**
CO4	Integrate ideas from various domains and implement them using blockchain technologyin different perspectives.	**PO5, PO6, PO12**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3
ETCA519A	Blockchain	3	3	3		2	2						2	3		2

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCS464A	Major Project	L	T	P	C
Version 1.0		-	-	-	6
Pre-requisites/Exposure	--				
Co-requisites	--				

The course is designed to provide an opportunity to students to demonstrate the ability to devise, select and use a range of methodologies and tools to the Chosen/Given project, applying the theoretical knowledge to a real life situation. Experiential Learning outside classroom through self-exploration, practical experience, Industry, field experience, live experience, research, design projects etc.

The learning process in the Project seeks out and focuses attention on many latent attributes, which do not surface in the normal class room situations. These experiential learning attributes through project includes Intellectual ability, Professional judgment and decision making ability, Inter-disciplinary approach, Skills for data handling, Ability in written and oral presentation, Sense of responsibility Developing professional Skills Application of theory, concepts in given industry /practical / field scenario.

Course Outcomes

On completion of this course, the students will be able to

CO1. Use applied scientific knowledge to identify and implement relevant principles of mathematics and computer science.
C02. Use the relevant tools necessary for engineering practice.
CO3. Define overall needs and constraints to solve a problem and develop/ design a prescribed engineering sub-system.
CO4. Communicate effectively and learn to be a team player.

Catalog Description

Students are expected make a project based on the latest advancements related to the parent branch of Engineering. Students may opt for an in-disciplinary project (if feasible).
The project may be a complete hardware or a combination of hardware and software under the guidance of a Supervisor from the Department. This is expected to provide a good training for the student(s) in technical aspects

Student will be continuously evaluated during the semester in form of Project ProgressSeminars. At the end of the semester, assessment of the research/project work of each student will bemade by the board of examiners including supervisors on the basis of a viva-voceexamination and the

report submitted by the student.

Course Content

The assignment to normally include:

1. Review and finalization of the Approach to the Problem relating to the assigned topic.
2. Preparing an Action Plan for conducting the investigation, including team work.
3. Detailed Analysis/Modelling/Simulation/Design/ProblemSolving/Experiment as needed.
4. Final development of product/process, testing, results, conclusions and future directions.
5. Preparing a report in the standard format for being evaluated by the Department.
6. Final projectpresentation before a Departmental Committee.

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination
Examination Scheme:

Components	Quiz	QuizI	Mid Term Exam	Presentation/ Assignment/ etc.	End Term Exam
Weightage (%)	**10**	**10**	**20**	**10**	**50**

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Use applied scientific knowledge to identify and implement relevant principles of mathematics and computer science.	**PO3**
CO2	Use the relevant tools necessary for engineering practice.	**PO5**
CO3	Define overall needs and constraints to solve a problem and develop/ design a prescribed engineering sub-system.	**PO3**
CO4	Communicate effectively and learn to be a team player.	**PO10**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO4	PO 5	PO 6	PO7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2	PSO3
ETCS 464A	Major Project			3		2					3					3

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCS422A	**Cloud Computing**	L	T	P	C
Version 1.0		4	0	0	4
Pre-requisites/Exposure	--				
Co-requisites	--				

Course Objectives

1. To provide students with the fundamentals and essentials of Cloud Computing.
2. To provide students a sound foundation of the Cloud Computing so that they are able to start using and adopting Cloud Computing services and tools in their real-life scenarios.
3. To enable students exploring some important cloud computing driven commercial systems and applications.
4. To expose the students to frontier areas of Cloud Computing and information systems, while providing sufficient foundations to enable further study and research.

Course Outcomes

On completion of this course, the students will be able to

CO1. Implement a public cloud instance using a public cloud service provider.

CO2. Explain the core concepts of the cloud computing paradigm: how and why this paradigm shift came about, the characteristics, advantages and challenges brought about by the various models and services in cloud computing.

CO3. Apply the fundamental concepts in data centres to understand the trade-offs in power, efficiency and cost.

CO4. Apply trust-based security model to different layers.

CO5. Develop a risk-management strategy for moving to the Cloud.

CO6. Describe big data and use cases from selected business domains.

CO7. Identify resource management fundamentals, i.e. resource abstraction, sharing and sandboxing and outline their role in managing infrastructure in cloud computing.

CO8. Analyze various cloud programming models and apply them to solve problems on the cloud.

Catalog Description

The course presents a top-down view of cloud computing, from applications and administration to programming and infrastructure. Its focus is on parallel programming techniques for cloud computing and large-scale distributed systems which form the cloud infrastructure. The topics include overview of cloud computing, cloud systems, parallel processing in the cloud, distributed storage systems, virtualization, security in the cloud, and multicore operating systems. Students will study state-of-the-art solutions for cloud computing developed by Google, Amazon, Microsoft, Yahoo, VMWare, etc. Students will also apply what they learn in one programming assignment and one project executed over Amazon Web Services.

Course Content

Unit I: **10 lecture hours**

Introduction: Cloud computing fundamentals, the role of networks in Cloud computing, Essential characteristics of Cloud computing, Cloud deployment model, Cloud service models, Multi-tenancy, Cloud cube model, Cloud economics and benefits, Cloud types and service scalability over the cloud, challenges in cloud NIST guidelines, Cloud economics and benefits, Cloud computing platforms - IaaS: Amazon EC2, PaaS: Google App Engine, Microsoft Azure, SaaS. Open Source platforms: OpenStack.

Unit II: **12 lecture hours**

Virtualization, Server, Storage and Networking: Virtualization concepts, types, Server virtualization, Storage virtualization, Storage services, Network virtualization, service virtualization, Virtualization management, Virtualization technologies and architectures, Internals of virtual machine, Measurement and profiling of virtualized applications. Hypervisors: KVM, Xen, Hyper V, VMware hypervisors and their features.

Unit III: **10 lecture hours**

Data in Cloud Computing: Relational databases, Cloud file systems: GFS and HDFS, BigTable, HBase and Dynamo. MapReduce and extensions: Parallel computing, the map-Reduce model, Parallel efficiency of MapReduce, Relational operations using Map-Reduce, Enterprise batch processing using MapReduce.

Cloud Security: Cloud security fundamentals, Vulnerability assessment tool for cloud, Privacy and Security in cloud. Cloud computing security architecture: General Issues, Trusted Cloud computing, Secure Execution Environments and Communications, Micro - architectures; Identity Management and Access control, Autonomic security, Security challenges: Virtualization security management - virtual threats, VM Security Recommendations, VM - Specific Security techniques, Secure Execution Environments and Communications in cloud.

Unit IV: **8 lecture hours**

Issues in Cloud Computing: Implementing real time application over cloud platform, Issues in Inter -cloud environments, QOS Issues in Cloud, Dependability, data migration, streaming in Cloud. Quality of Service (QoS) monitoring in a Cloud computing environment. Cloud Middleware. Mobile Cloud Computing. Inter Cloud issues. A grid of clouds, Sky computing, load

balancing, resource optimization, resource dynamic reconfiguration, Monitoring in Cloud

Text Books

1. Cloud Computing, Dr. Kumar Saurabh, Wiley Publication

Reference Books/Materials

1. Cloud computing – Automated virtualized data center, Venkata Josyula, CISCO Press

2. Cloud and virtual data storage networking, Greg Schulr CRC Press

3. Handbook of Cloud Computing, Borko Furht, Springer

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination
Examination Scheme:

Components	Quiz	Attendance	Mid Term Exam	Presentation/ Assignment/ etc.	End Term Exam
Weightage (%)	10	10	20	10	50

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Implement a public cloud instance using a public cloud service provider.	**PO5**
CO2	Explain the core concepts of the cloud computing paradigm: how and why this paradigm shift came about, the characteristics, advantages and challenges brought about by the various models and services in cloud computing.	**PO1**
CO3	Apply the fundamental concepts in data centres to understand the trade-offs in power, efficiency and cost.	**PO4**
CO4	Apply trust-based security model to different layers.	**PO5**
CO5	Develop a risk-management strategy for moving to the Cloud.	**PO2**

CO6	Describe big data and use cases from selected business domains.	**PO3**
CO7	Identify resource management fundamentals, i.e. resource abstraction, sharing and sandboxing and outline their role in managing infrastructure in cloud computing.	**PO3**
CO8	Analyze various cloud programming models and apply them to solve problems on the cloud.	**PO9**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2	PSO3
ETCS422A	Cloud Computing	2	3	3	2	3				3				3		

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCA 362A	Cloud Computing Lab	L	T	P	C
Version 1.0		0	0	2	1
Pre-requisites/Exposure	Practical learning				
Co-requisites	--				

Course Objectives

1. Define & implement Virtualization using different types of Hypervisors
2. Describe steps to perform on demand application delivery
3. Examine the installation and configuration of Open stack cloud
4. Analyze and understand the functioning of different components involved in Amazon web services cloud platform.
5. Describe the functioning of Platform as a Service
6. Design & Synthesize Storage as a service using own Cloud

Course Outcomes

On completion of this course, the students will be able to

CO1. Implement a public cloud instance using a public cloud service provider.

CO2. Explain the core concepts of the cloud computing paradigm: how and why this paradigm shift came about, the characteristics, advantages and challenges brought about by the various models and services in cloud computing.

CO3. Apply the fundamental concepts in data centres to understand the trade-offs in power, efficiency and cost.

CO4. Apply trust-based security model to different layers.

CO5. Develop a risk-management strategy for moving to the Cloud.

CO6. Describe big data and use cases from selected business domains.

CO7. Identify resource management fundamentals, i.e. resource abstraction, sharing and sandboxing and outline their role in managing infrastructure in cloud computing.

CO8. Analyze various cloud programming models and apply them to solve problems on the cloud.

Catalog Description

This course is designed to introduce the concepts of Cloud Computing as a new computing paradigm. The students will have an opportunity to explore the Cloud Computing various terminology, concepts, principles and applications. This course provides a hands-on comprehensive study of Cloud concepts and capabilities across the various Cloud service models including Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS). A variety of real case studies and existing in market cloud- based tools will be identified and studied in order to provide students with a close overview to Cloud Computing applications.

Course Content

1	Development of applications on Google app engine.	**4 lab hours**
2	Case study of private Cloud setup through OpenStack	**4 lab hours**
3	Case study of private Cloud setup through CloudStack	**4 lab hours**
4	Case study of XEN/VMware/KVM hypervisor	**4 lab hours**
5	Case study of Amazon ec2.	**4 lab hours**

Modes of Evaluation: Quiz/Oral practical oral exam/presentation/projects/Practical Examination

Examination Scheme:

Components	**Quiz**	**Attendance**	**Mid Term Exam**	**Presentation/ Assignment/ etc.**	**End Term Exam**
Weightage (%)	**10**	**10**	**20**	**10**	**50**

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Implement a public cloud instance using a public cloud service provider.	**PO5**
CO2	Explain the core concepts of the cloud computing paradigm: how and why this paradigm shift came about, the characteristics, advantages and challenges brought about by the various models and services in cloud computing.	**PO1**

CO3	Apply the fundamental concepts in data centres to understand the trade-offs in power, efficiency and cost.	**PO4**
CO4	Apply trust-based security model to different layers.	**PO5**
CO5	Develop a risk-management strategy for moving to the Cloud.	**PO2**
CO6	Describe big data and use cases from selected business domains.	**PO3**
CO7	Identify resource management fundamentals, i.e. resource abstraction, sharing and sandboxing and outline their role in managing infrastructure in cloud computing.	**PO3**
CO8	Analyze various cloud programming models and apply them to solve problems on the cloud.	**PO9**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS O1	PS O2	PS O3

ETCA362 A	Cloud Computing Lab	2	3	3	2	3				3				3		

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCS 424A	**Data Warehouse and Data Mining**	L	T	P	C
Version 1.0		4	0	0	4
Pre-requisites/Exposure	Basic Database concepts, Query tools				
Co-requisites	--				

Course Objectives

1. Be familiar with mathematical foundations of data mining tools.

2. Understand and implement classical models and algorithms in data warehouses and data mining

3. Characterize the kinds of patterns that can be discovered by association rule mining, classification and clustering.

4. Master data mining techniques in various applications like social, scientific and environmental context.

5. Develop skill in selecting the appropriate data mining algorithm for solving practical problems.

Course Outcomes

On completion of this course, the students will be able to:

CO1. Understand the functionality of the various data mining and data warehousing component

CO2. Appreciate the strengths and limitations of various data mining and data warehousing models

CO3. Explain the analyzing techniques of various data

CO4. Describe different methodologies used in data mining and data ware housing

CO5. Compare different approaches of data ware housing and data mining with various technologies

Catalog Description

This course will introduce the concepts of data ware house and data mining, which gives a complete description about the principles, used, architectures, applications, design and implementation of data mining and data ware housing concepts.

Course Content

Unit I: **10 lecture hours**

Introduction: Evolution Of Data Warehousing (Historical Context), The Data Warehouse - a Brief Overview, Characteristics, Operational Database Systems and Data Warehouse(OLTP & OLAP), Data Marts, Metadata.

Principles of Data Warehousing(Architecture and Design Techniques): System Processes, Data Warehousing Components, Architecture for a Warehouse, Three-tier Data Warehouse Architecture, Steps for the design and construction of Data Warehouses, Conceptual Data Architecture, Logical Architectures, Design Techniques.

Unit II: **10 lecture hours**

Multidimensional Data Models: Types of Data and Their Uses, From Tables and Spreadsheets to Data Cubes, Identifying Facts and Dimensions, Fact Tables, Designing Fact Tables, Designing Dimension Table, Data Warehouse Schemas- STAR Schema, Snowflake Schema, OLAP, OLAP Operations**,** Hypercube**,** ROLAP, MOLAP, From Data warehousing to Data Mining, Data warehouse Usage

Unit III: **10 lecture hours**

Data Mining: Motivation, Importance, Knowledge Discovery Process (KDD), KDD and Data Mining, Data Mining vs. Query Tools, Kind of Data, Data preprocessing, Functionalities, Interesting Patterns, Classification of data mining systems, Major issues.

Unit IV: **10 lecture hours**

Classification and Prediction: Classification & Prediction, Issues Regarding Classification & Prediction, Classification by Decision Tree Induction, Bayesian Classification, Classification by Back Propagation, Classification Parameters.

Cluster Analysis: Types of Data in Cluster Analysis, Partitioning Method, Hierarchical Method, Density Based Method, Grid Based Method, Model Based Clustering Method, Outlier Analysis.

Mining Association Rules: Association Rule Mining, Market Basket Analysis, Types of Association Rules, Methods for Mining Association

Text Books

Kamber and Han, "Data Mining Concepts and Techniques", Hartcourt India P. Ltd

Reference Books/Materials

1. W. H. Inmon, "Building the operational data store", 2nd Ed., John Wiley.
2. Paul Raj Poonia, "Fundamentals of Data Warehousing", John Wiley & Sons.
3. Sam Anahony, "Data Warehousing in the real world: A practical guide for building decision support systems", John Wiley.

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination
Examination Scheme:

Components	Quiz	Attendance	Mid Term Exam	Presentation/ Assignment/ etc.	End Term Exam
Weightage (%)	10	10	20	10	50

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Understand the functionality of the various data mining and data warehousing component	**PO1**

CO2	Appreciate the strengths and limitations of various data mining and data warehousing models	**PO1**
CO3	Explain the analyzing techniques of various data	**PO2**
CO4	Describe different methodologies used in data mining and data ware housing	**PO2**
CO5	Compare different approaches of data ware housing and data mining with various technologies	**PO4, PO5**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behavior	Knowledge
Course Code	Course Title	PO1	PO2	PO 3	PO4	PO5	PO 6	PO 7	PO 8	PO9	PO 10	PO1 1	PO 12	PSO1	PSO2	PSO3
ETCS 424A	Data warehouse and data mining	3	3		3	3								3	2	2

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCS463A	Data Warehousing and Data MiningLab	L	T	P	C
Version 1.0		0	0	2	1
Pre-requisites/Exposure	Basic Database concepts, Query tools				
Co-requisites	--				

Course Objectives

1. Be familiar with mathematical foundations of data mining tools.

2. Understand and implement classical models and algorithms in data warehouses and data mining

3. Characterize the kinds of patterns that can be discovered by association rule mining, classification and clustering.

4. Master data mining techniques in various applications like social, scientific and environmental context.

5. Develop skill in selecting the appropriate data mining algorithm for solving practical problems.

Course Outcomes

On completion of this course, the students will be able to:

CO1. Able to get the acquaintance to WEKA tool

CO2. Competent to preprocess the data for mining

CO3. Proficient in generating association rules

CO4. Able to build various classification models

CO5. Able to realize clusters from the available data

Catalog Description

The main objective of this lab is to impart the knowledge on how to implement classical models and algorithms in data warehousing and data mining and to characterize the kinds of patterns that can be discovered by association rule mining, classification and clustering. At the end, the course provides a comparison of different conceptions of data mining.

List of Experiments (Indicative)

1	Demonstration of data pre-processing on datasets	**2 lab hours**
2	To list all the categorical (or nominal) attributes and the real valued attributes	**4 lab hours**
3	Create a data classification model using decision tree	**4 lab hours**
4	Create a data classification model using Naive Bayes	**2 lab hours**
5	Create a data classification model using rule based classifies	**2 lab hours**
6	Create a data classification model using statistical classifiers.	**4 lab hours**
7	Create a data classification model using neural networks.	**4 lab hours**
8	Create a data classification model	**4 lab hours**
9	Demonstrate the working of k-means algorithm for clustering the data.	**4 lab hours**
10	Create a clustering model using hierarchical clustering algorithm.	**2 lab hours**

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination
Examination Scheme:

Components	**Quiz**	**Attendance**	**Mid Term Exam**	**Presentation/ Assignment/ etc.**	**End Term Exam**
Weightage (%)	**10**	**10**	**20**	**10**	**50**

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Able to get **the** acquaintance to WEKA tool	**PO5**
CO2	Competent to preprocess the data for mining	**PO2**
CO3	Proficient in generating association rules	**PO4**

CO4	Able to build various classification models	**PO3**
CO5	Able to realize clusters from the available data	**PO4**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behavior	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO4	PO 5	PO 6	PO7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2	PSO3
ETCS 463A	Data warehouse and data mining Lab		2	3	3	3								3	2	2

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCS421A	Internet of Things	L	T	P	C
Version 1.0		4	0	0	4
Pre-requisites/Exposure	Sensors, System Integration				
Co-requisites	Cloud and Network Security				

Course Objectives

The objective of this course is to impart necessary and practical knowledge of components of Internet of Things and develop skills required to build real-time IoT based projects

Course Outcomes

On completion of this course, the students will be able to

CO1. Understand IoT and its hardware and software components
CO2. Interface I/O devices, sensors and communication mobiles
CO3. Remotely monitor data and control devices
CO4. Develop real life IoT based projects

Catalog Description

The Internet of Things (IoT) is everywhere. It provides advanced data collection, connectivity, and analysis of information collected by computers everywhere—taking the concepts of Machine-to-Machine communication farther than ever before. This course gives a foundation in the Internet of Things, including the components, tools, and analysis by teaching the concepts behind the IoT and a look at real-world solutions.

Course Content

Unit I: 8 lecture hours

Introduction to IoT: Defining IoT, Characteristics of IoT, Physical design of IoT, Logical design of IoT, Functional blocks of IoT, Communication models & APIs. Machine to Machine, Difference between IoT and M2M, Software Define Network

Unit II: 9 lecture hours

Network and Communication Aspects: Wireless medium access issues, MAC protocol survey, Survey routing protocols, Sensor deployment & Node discovery, Data aggregation & dissemination.

Unit III: **10 lecture hours**

Challenges in IoT: Design challenges, Development challenges, Security challenges, other challenges. Home automation, Industry applications, Surveillance applications, Other IoT applications

Unit IV: **12 lecture hours**

Developing IoT's: Input/output Programming: Introduction to different IoT tools, Developing applications through IoT tools, Developing sensor based application through embedded system platform, Implementing IoT concepts with python

Text Books

1. Vijay Madisetti, ArshdeepBahga, "Internet of Things: A Hands-On Approach"

2. WaltenegusDargie, Christian Poellabauer, "Fundamentals of Wireless Sensor Networks: Theory and Practice"

Modes of Evaluation: Quiz/Assignment/ presentation/ extempore/ Written Examination
Examination Scheme:

Components	Quiz	Attendance	Mid Term Exam	Presentation/ Assignment/ etc.	End Term Exam
Weightage (%)	10	10	20	10	50

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	Mapped Program Outcomes
CO1	Understand IoT and its hardware and software components	PO2
CO2	Interface I/O devices, sensors and communication mobile.	PO1
CO3	Remotely monitor data and control devices	PO4
CO4	Develop real life IoT based projects	PO3

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2	PSO3
ETCS421A	Internet of Things	2	3	3	3									3		3

1=weakly mapped
2= moderately mapped
3=strongly mapped

ETCS457A	**Internet of Things Lab**	L	T	P	C
Version 1.0		0	0	2	1
Pre-requisites/Exposure	Sensors, System Integration				
Co-requisites	Cloud and Network Security				

Course Objectives

The objective of this course is to impart necessary and practical knowledge of components of Internet of Things and develop skills required to build real-time IoT based projects

Course Outcomes

On completion of this course, the students will be able to

CO1. Understand IoT and its hardware and software components
CO2. Interface I/O, sensors and communication mobiles
CO3. Remotely monitor data and control devices
CO4. Develop real life IoT based projects

Catalog Description

This course complements ETCS 480A. This course gives a foundation in the Internet of Things, including the components, tools, and analysis by teaching the concepts behind the IoT and a look at real-world solutions.

List of Experiments (Indicative)

1	Start Raspberry Pi and try various Linux commands in command terminal window	**2 lab hours**
2	Read your name and print Hello message with name.	**2 lab hours**
3	Read two numbers and print their sum, difference, product and division.	
4	Word and character count of a given string	
5	Area of a given shape (rectangle, triangle and circle) reading shape and appropriate values from standard input	**2 lab hours**
6	Print a name 'n' times, where name and n are read from standard input, using for and while loops.	
7	Handle Divided by Zero Exception.	
8	Print current time for 10 times with an interval of 10 seconds.	**2 lab hours**
9	Read a file line by line and print the word count of each line.	
10	To interface LED/Buzzer with Arduino/ Raspberry pi and write a program to turn ONLED for 1 secaftere very 2 seconds.	**2 lab hours**
11	Switch on a relay at a given time using cron, where the relay's contact terminals	**2 lab hours**

	are connected to a load.	
12	To install My SQL data base on Raspberry Pi and perform basic SQL queries.	**2 lab hours**
13	Write a program on Arduino /Raspberry Pi to publish temperature data to MQTT broker.	**2 lab hours**
14	WriteaprogramonArduino/RaspberryPitosubscribetoMQTTbrokerfortemperature dataandprint it.	**2 lab hours**
15	WriteaprogramtocreateTCPserveronArduino/RaspberryPiandrespondwithhumidity data to TCP client when requested..	**3 lab hours**

Modes of Evaluation: Quiz/Oral practical oral exam/presentation/projects/Practical Examination

Examination Scheme:

Components	**Quiz**	**Attendance**	**Mid Term Exam**	**Presentation/ Assignment/ etc.**	**End Term Exam**
Weightage (%)	**10**	**10**	**20**	**10**	**50**

Relationship between the Course Outcomes (COs) and Program Outcomes (POs)

Mapping between COs and POs		
	Course Outcomes (COs)	**Mapped Program Outcomes**
CO1	Understand IoT and its hardware and software components	**PO2**
CO2	Interface I/O devices, sensors and communication mobile.	**PO1**
CO3	Remotely monitor data and control devices	**PO4**
CO4	Develop real life IoT based projects	**PO3**

		Engineering Knowledge	Problem analysis	Design/development of solutions	Conduct investigations of complex problems	Modern tool usage	The engineer and society	Environment and sustainability	Ethics	Individual or team work	Communication	Project management and finance	Life-long Learning	Employability	Ethics and Behaviour	Knowledge
Course Code	Course Title	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2	PSO3
ETCS482A	Internet of Things Lab	2	3	3	3									3		3

1=weakly mapped
2= moderately mapped
3=strongly mapped