K.R. MANGALAM UNIVERSITY
THE COMPLETE WORLD OF EDUCATION

SCHOOL OF ENGINEERING & TECHNOLOGY

Bachelor of Science (Honours) Computer Science
B.Sc. (H) Computer Science

Program Code: 72

2019-2022

Approved in the 20th Meeting of

Academic Council Held on 16 July

2019

Academic Council
Member Secretary
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 & TECHNOLOGY

Bachelor of Science (Honours) Computer Science
B.Sc. (H) Computer Science

Program Code: 72

2019-2022

Approved in the 20th Meeting of

Academic Council Held on 16 July

2019

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 programmes 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.

Programs 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 Programme is aimed at developing a sound knowledge and understanding of concepts in key areas of Operating System, Database System Concepts, Computer Networks, Web Technologies, Java Programming, Python Programming, etc. An initiative to make the teaching-learning framework better and enhance the student learning outcomes, the School has taken a thoughtful step by introducing the concept of Choice Based Credits System (CBCS) system.

Program Duration: 3 Years (6 Semesters)

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/Computer Science/Informatics Practice/ on as one of the subjects and with an overall minimum aggregate of 50% or more.

Career Options

For B.Sc. (H) (Computer Science):- System integrator, Hardware Designer, Logic Designer, Systems Analyst, System Administrator, Computer programmer, Computer Scientist.

Class Timings

The classes will be held from Monday to Friday from 9.10 a.m to 4.10 p.m.

Scheme of Studies and Syllabi

For B.Sc.(H) Computer Science program scheme attached in Annexure A1.

The syllabi of all courses for first year for all the programmes offered by the school are given in the following pages. These are arranged as: (a) common course (b) degree specific course, in numeric order of the last three digits of the course code.

For each course, the first line contains; Course Code, Title and Credits (C) of the course. This is followed by detailed syllabi.

Three Years B.Sc. (H) Computer Science Programme At A Glance

	Sem I	Sem II	Sem III	Sem IV	Sem V	Sem VI	Total
Course	9	9	9	8	9	8	**52**
Credit	24	25	26	22	20	21	**138**

Scheme as per Choice Based Credit System (CBCS)

SOET — **B.SC.(H) COMPUTER SCIENCE 2019-23 (SCHEME OF STUDIES)**

YEAR	ODD SEMESTER								EVEN SEMESTER							
	SNO	Category	COURSE CODE	COURSE TITLE	L	T	P	C	SNO	Category	COURSE CODE	COURSE TITLE	L	T	P	C
FIRST	1	SE	ETME101A	BASICS OF MECHANICAL ENGINEERING	3	1	-	4	1	SE	ETPH120A	FUNDAMENTALS OF PHYSICS-I	3	1	-	4
	2	PEC		OPEN ELECTIVE- I	4	-	-	4	2	PEC		OPEN ELECTIVE-II	4	-	-	4
	3	SE	ETEL 101A	COMMUNICATION SKILLS	4	-	-	4	3	SE	ETEL 402A	SOFT SKILLS AND PERSONALITY DEVELOPMENT	3	-	-	3
	4	EMP	ETCS103A	PROGRAMMING FOR PROBLEM SOLVING	3	1	-	4	4	SE	ETCH 125A	ENVIRONMENTAL STUDIES	3	-	-	3
	5	EMP	ETCS521A	COMPUTER CENTRE PLANNING & ESTABLISHMENT	3	1	-	4	5	EMP	ETCS 316A	WEB TECHNOLOGIES	3	1	-	4
	6	SE	ETME151A	BASICS OF MECHANICAL ENGINEERING LAB	-	-	2	1	6	EMP	ETCS112A	OBJECT ORIENTED PROGRAMMING	3	1	-	4

	7	SE	ETEL171A	COMMUNICATION SKILLS LAB	-	-	2	1		7	SE	ETPH158A	FUNDAMENTALS OF PHYSICS-I LAB	-	-	2	1
	8	EMP	ETCS 153A	PROGRAMMING FOR PROBLEM SOLVING LAB	-	-	2	1		8	EMP	ETCA164A	WEB TECHNOLOGIES LAB	-	-	2	1
	9	EMP	ETCS565A	COMPUTER CENTRE PLANNING & ESTABLISHMENT LAB	-	-	2	1		9	EMP	ETCS166A	OBJECT ORIENTED PROGRAMMING LAB	-	-	2	1
				TOTAL	**17**	**2**	**8**	**24**					**TOTAL**	**19**	**3**	**6**	**25**
SECOND	1	SE	ETEC 210A	DIGITAL ELECTRONICS	3	1	-	4		1	SE	ETPH217A	FUNDAMENTALS OF PHYSICS-II	4	1	-	4
	2	SE	BSMA217A	REAL ANALYSIS	4	-	-	4		2	SE	ETMA137A	ALGEBRA	4	-	-	4
	3	SE	ETMA208A	NUMERICAL ANALYSIS	4	-	-	4		3	EMP	ETCS 220A	ANALYSIS AND DESIGN OF ALGORITHMS	3	1	-	4
	4	EMP	ETCS219A	FOUNDATION OF COMPUTER SYSTEMS	3	1	-	4		4	SE	ETMC 226A	FUNDAMENTALS OF MANAGEMENT	3	-	-	3
	5	EMP	ETCS217A	DATA STRUCTURES	3	1	-	4		5	EMP	ETCS307A	DATABASE MANAGEMENT SYSTEMS	3	1	-	4
	6	EMP	ETCS 257A	DATA STRUCTURES LAB	-	-	2	1		6	SE	ETPH257A	FUNDAMENTALS OF PHYSICS-II Lab	-	-	2	1
	7	EMP	ETEC 256A	DIGITAL ELECTRONICS LAB	-	-	2	1		7	EMP	ETCS 355A	DATABASE MANAGEMENT SYSTEMS LAB	-	-	2	1
	8	SE	BSMA351A	NUMERICAL ANALYSIS LAB	0	0	2	1		8	EMP	ETCA365A	LINUX ENVIRONMENT LAB	-	-	2	1
	9	SE	ETDM301A	Disaster Management	3	-	-	3									
				TOTAL	**20**	**3**	**6**	**26**					**TOTAL**	**17**	**3**	**6**	**22**

THIRD	1	EMP	ETCS304A	COMPUTER NETWORKS	3	1		4	1	EMP	ETCS506A	PYTHON PROGRAMMING	3	1	-	4
	2	EMP	ETCS211A	OPERATING SYSTEMS	3	1	-	4	2	EMP	ETCS202A	SOFTWARE ENGINEERING	3	1	-	4
	3	EMP	ETCS 206A	COMPUTER GRAPHICS	3	1		4	3	EMP	ETCA324A	.Net FRAMEWORK	3	1	-	4
	4	SE	BSMA326A	OPERATIONAL RESEARCH	4	-	-	4	4	SE	ETCS505A	TECHNICAL WRITING	3	-	-	3
	5	EMP	ETCS365A	COMPUTER NETWORKS LAB	-	-	2	1	5	EMP	ETCA364A	.Net FRAMEWORK LAB	-	-	2	1
	6	EMP	ETCS 255A	OPERATING SYSTEMS LAB	-	-	2	1	6	EMP	ETCS555A	PYTHON PROGRAMMING LAB	-	-	2	1
	7	EMP	ETCS258A	COMPUTER GRAPHICS LAB	-	-	2	1	7	EMP	ETCS252A	SOFTWARE ENGINEERING LAB	-	-	2	1
	8	SE	ETCS507A	SEMINAR	-	-	2	1	8	SE	ETCA368A	MAJOR PROJECT	-	-	6	3
				TOTAL	**15**	**3**	**8**	**20**				**TOTAL**	**12**	**3**	**12**	**21**
												TOTAL CREDITS	**138**			

SEMESTER I

ETME 101A	BASICS OF MECHANICAL ENGINEERING	C
		4

Course Overview:

This is one of the core subjects that introduces the student to the study of various mechanical engineering concepts and prepares the student for further studies and better understanding of engineering subjects like Engineering Thermodynamics, strength of materials and theory of machines, etc.

Objectives and Expected Outcomes

The subject expects students to achieve the following objectives.

- To analyze, design and improve practical thermal and/or mechanical systems.
- To give students practice in applying their knowledge of mathematics, science, and engineering and to expand this knowledge into the vast area of mechanical engineering.
- To enhance students' ability to design by requiring the solution of open ended problems.
- To prepare the students for higher level courses such as courses in Mechanics of Solids, Thermodynamics, Manufacturing, etc.

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

- Know the basics of machine tool and their material properties.
- Understand the basic concepts of thermodynamics and Refrigeration.
- Get the knowledge of application of hydraulic turbines and pumps in various fields.
- Know various Power Transmission Methods and Devices.
- Understand the concept of Stress & Strain which is useful in various streams of engineering.

Unit I

Introduction to Machine Tools and Commonly used Machine Tools in a Workshop: Lathe, Shaper, Planer, Milling, Drilling, Slotter, Introduction to Metal Cutting.

Basic concept of thermodynamics: Introduction, States, Work, Heat, Temperature, Zeroth, 1st, 2^{nd} and 3^{rd} law of thermodynamics, Concept of internal energy, enthalpy and entropy. Problems Properties of Steam & Steam Generator Formation of steam at constant pressure, Thermodynamic properties of Steam, Use of steam tables, Measurement of dryness fraction by throttling calorimeter.

Unit II

Refrigeration & Air-conditioning: Introduction to refrigeration and air -conditioning, Rating of refrigeration machines, Coefficient of performance, Simple refrigeration vapor compression cycle,

Psychrometric charts and its use, Human comforts.

Hydraulic Turbines & Pumps: Introduction, Classification, Construction details and working of Pelton, Francis and Kaplan turbines, Specific speed and selection of turbines, Classification of water pumps and their working.

Unit III

Power Transmission Methods and Devices: Introduction to Power transmission, Belt, Rope, Chain and Gear drive, Types and functioning of clutches

Stresses and Strains: Introduction, Concept & types of Stresses and strains, Poison's ratio, stresses and strains in simple and compound bars under axial, flexure & torsional loading, Stress- strain diagrams, Hooks law, Elastic constants & their relationships.

Unit IV

Introduction to Manufacturing Systems: Fundamentals of Numerical Control (NC), Advantage of NC systems, Classifications of NC, Comparison of NC and CNC

Text Books:

1. Elements of Mechanical Engineering – R.K.Rajput Lakmi Pub., Delhi
2. Elements of Mechanical Engineering – D.S.Kumar, S.K. Kataria and Sons
3. Engineering Thermodynamics- P.K.Nag TMH, New Delhi
4. Refrigeration & Air-conditioning – Arora & Domkundwar, Dhanpat rai & co.pvt ltd
5. Workshop Technology Vol.I & II - Hazra & Chaudhary, Asian Book Comp., New Delhi.
6. Process and Materials of Manufacture -- Lindberg, R.A. Prentice Hall of India, New Delhi.
7. Principles of Manufacturing Materials and Processes - Campbell, J.S.- McGraw- Hill

Reference Books:

1. Strength of Materials – Popov, Pub. PHI, New Delhi.
2. Hydraulic Machines – Jagdish Lal, Pub. Metropolitan, Allahabad.
3. Strength of Materials - G.H. Ryder, Pub. ELBS.

OPEN ELECTIVE - I	C
	6

There are three open electives offered by other departments / schools in first semester. The department will permit student to opt one open elective based on choice of student and consent of the course advisor.

S. No.	Course Code	Course Title
1	IIIT101	Harnessing the Power of the Web as a Knowledge Device
2	IIIT107	Art of Logic & Programming
3	ETCE101	Fundamentals of Civil Engineering

ETEL 101A	COMMUNICATION SKILLS	C
		4

Course Overview:

The world is shrinking into a global village and therefore, communication skills in English have emerged as a major means of empowerment and human resource development. Many professionals fail to make an impact on the global market as they lack the required communicative competence. The course will augment comprehension skills, enhance vocabulary, and enable to acquire impressive writing skills, assist correspondence with others effectively, aid in understanding the non-verbal cues and enrich skills in spoken English through a variety of teaching techniques. The course will be instrumental in acquiring proficiency both in spoken and oral language.

Objectives and Expected Outcomes

The course will help the learners to focus on communication activities in functional and situational contexts as well as enhance the four language skills of reading, writing, listening and speaking through real-life and professional situations. It will build confidence among the students and encourage them to speak fluently. Through practical learning and delivery, the learners will be able to identify their areas of strengths and weaknesses and improvise their personality and soft skills. The learners will be able to strengthen and broaden their communication skills through various insightful ways.

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.

UNIT I

Introduction to Communication: Meaning, Forms & Types of Communication; Process of Communication; Principles of Effective Communication/7Cs, Barriers in Communication; Literature: A Bird Came Down the Walk by Emily Dickinson

UNIT II

Essentials of Grammar: Parts of Speech: Noun, Pronoun, Adjective, Verb, Adverb, Preposition, Conjunction, Interjection; Using tenses; Articles; Types of sentences; Reported Speech; Punctuation; Literature: Stopping by Woods on A Snowy Evening by Robert Frost

UNIT III

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); Literature: The Gift of Magi by O'Henry

UNIT IV

Personality Development: Etiquette & Manners; Leadership; Inter & intra personal skills; Attitude, Self-esteem & Self-reliance; Public Speaking; Body Language: Posture, Gesture, Eye Contact, Facial Expressions;

Presentation Skills/ Techniques; Literature: My Prayer to Thee by Rabindranath Tagore;

TEXT BOOK:

Kumar, Sanjay and Pushplata. Communication Skills. Oxford University Press.

REFERENCE BOOKS / SITES:

1. Tickoo, M.L, Subramanian A. E. and Subramaniam P.R. Intermediate Grammar, Usage and Composition. Orient Blackswan.

2. Mitra, Barun K. Personality Development and Soft Skills. Oxford University Press.

3. “Best Poems”, http://100.best-poems.net/. 20 July 2016.

4. “Classic English Short Stories”, http://www.eastoftheweb.com/short-stories/Collections/ClasEngl.shtml, 20 July 2016.

ETCS103A	PROGRAMMING FOR PROBLEM SOLVING	C
		4

Course Overview:

Computer software plays an important role in our daily lives: Our mobile phones, laptop computers, online banking, Internet applications such as YouTube, video games and movies, cars, and almost all aspects of daily life are touched by software. In your personal and professional life you will utilize computer software. It is also likely that you will select, or even influence the design of, software that is used in your professional or personal life. This thematic sequence will give you a deep understanding of how software works and is created, its limitations, and its potential. You will be able to read software and therefore be able to make informed decisions when selecting or participating in the design of business, scientific, or information systems that utilize computer software. 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

Objectives and Expected Outcomes:

The student will learn

- To formulate simple algorithms for arithmetic and logical problems.
- To translate the algorithms to programs (in C language).
- To test and execute the programs and correct syntax and logical errors.
- To implement conditional branching, iteration and recursion.
- To decompose a problem into functions and synthesize a complete program using divide and conquer approach.
- To use arrays, pointers and structures to formulate algorithms and programs.
- To apply programming to solve matrix addition and multiplication problems and searching and sorting problems.
- To apply programming to solve simple numerical method problems, namely rot finding of function, differentiation of function and simple integration.

UNIT I

Introduction to Programming: Introduction to components of a computer system (disks, memory, processor, where a program is stored and executed, operating system, compilers etc.)

Idea of Algorithm: steps to solve logical and numerical problems. Representation of Algorithm:

Flowchart / Pseudocode with examples.

From algorithms to programs; source code, variables (with data types) variables and memory locations, Syntax and Logical Errors in compilation, object and executable code- Arithmetic expressions and precedence

UNIT II

Conditional Branching and Loops, Writing and evaluation of conditionals and consequent branching, Iteration and loops.

Arrays: Arrays (1-D, 2-D), Character arrays and Strings

UNIT III

Basic Algorithms: Searching, Basic Sorting Algorithms (Bubble, Insertion and Selection), Finding roots of equations, notion of order of complexity through example programs (no formal definition required)

Function: Functions (including using built in libraries), Parameter passing in functions, call by value, passing arrays to functions: idea of call by reference.

Recursion: Recursion, as a different way of solving problems. Example programs, such as Finding Factorial, Fibonacci series, Ackerman function, Quick sort or Merge sort.

UNIT IV

Structure**:** Structures, Defining structures and Array of Structures

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

Suggested Text Books

1. Byron Gottfried, Schaum's Outline of Programming with C, McGraw-Hill
2. E. Balaguruswamy, Programming in ANSI C, Tata McGraw-Hill

ETCS521A	COMPUTER CENTRE PLANNING & ESTABLISHMENT	C
		4

Course Overview:

Computing and programming is essential to leverage the technical skills of a student. These techniques equip the students with know-how of the latest technologies and reduce considerable time in solving problems. The course of Information Technology Fundamentals has become essentially the present age of computer technology and information, as the applications of information technology can be found in all aspects of our lives.

Objectives and Expected Outcomes:

The main objective is to introduce IT in a simple language to all undergraduate students, regardless of their specialization. It will help them to pursue specialized programs leading to technical and professional careers and certifications in the IT industry. The focus of the subject is on introducing skills relating to IT basics, computer applications, programming, interactive media, Internet basics, etc. At the end of this course, students should be able to

- Understand basic concepts and terminology of information technology.
- Have a basic understanding of personal computers and their operations.
- Understand the process of algorithm development and documentation

UNIT – I

Introduction to Computers:

The evolution of computers: Computer Generation from First Generation to Fifth Generation. Classifications of Computers: Micro, Mini, Mainframe and super computers, Distributed Computer System, Parallel Computers.
Computer Hardware: Major Components of a digital computer, Block Diagram of a computer Input devices, Output Device. Computer Memory: Memory Cell, Overview of Memory Organization, Primary Memory: RAM & ROM, Secondary memory: Magnetic tapes, Magnetic disk, CD-ROM, DVD.

UNIT – II

Introduction to System Software and Operating System:

Computer Software: Machine language, assembly language, high-level languages, fourth generation language, assemblers, compilers, interpreters, linkers, loaders.
Operating System concepts: different types of operating systems, functions of operating system, concept of multiprogramming, multitasking, multithreading, multiprocessing, time-sharing, real time, single-user & multi-user operating system.

UNIT – III

Programming Concepts & Techniques:

Algorithms, flow chart, decision tables, pseudo code, characteristics of a good programming language, Planning the Computer Program: Concept of problem solving, Problem definition, Program design, Debugging, Types of

errors in programming, Documentation.

Structured programming concepts, Programming methodologies viz. top-down and bottom-up programming, Advantages and disadvantages of Structured programming.

UNIT – IV

Computer Networks & The Internet:

Basic elements of a communication system, Data transmission modes, Data transmission media, Network topologies, Network Types (LAN, WAN and MAN), Client and Servers, Intranet, Extranet.

Internet: Terminology related to Internet: Protocols, TCP/IP, HTTP, Internet addressing, Domain Names, DNS, URL, World Wide Web. Overview of various services on Internet: Webservers, E-mail, FTP, Telnet.

TEXT BOOKS

1. P. K. Sinha & Priti Sinha , “Computer Fundamentals”, BPB Publications.

2. Anita Goel “Computer Fundamentals”, Pearson.

REFERENCE BOOKS

1. B.Ram Computer fundamentals Architecture and Organization,New Age Intl.

2. Alex Leon & Mathews Leon, “Introduction to Computers”, Vikas Publishing .

3. Norton Peter, “Introduction to computers”, TMH.

4. Vikas Gupta, “Comdex Computer Kit”, Wiley Dreamtech, Delhi.

ETME 151A	BASICS OF MECHANICAL ENGINEERING LAB	C
		1

Course Overview:

This is one of the core lab subjects that introduces the student to the study of various mechanical engineering concepts and prepares the student for further studies and better understanding of engineering subjects like Engineering Thermodynamics, strength of materials and theory of machines, etc. through practical exposure.

Objectives and Expected Outcomes:

- To calculate the Mechanical Advantage, Velocity Ratio and Efficiency of Single Start & Double Start Worm & Worm Wheel, Differential Wheel & Axle.
- To study simple screw jack and compound screw jack and determine their efficiency.
- To verify the law of Moments using Parallel Force apparatus. (simply supported type)
- To evaluate the co-efficient of friction between wood and various surface (like Leather, Wood, Aluminum) on an inclined plane.
- To Study Two-Stroke & Four-Stroke Diesel Engines and Petrol Engines.
- To Study the vapor compression Refrigeration System and Window Room Air Conditioner.
- To study the constructional features and working of Pelton wheel Turbine, Francis Turbine and Kaplan Turbine, etc.

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

- Understand the Mechanical Advantage, Velocity Ratio and Efficiency of various systems.
- Understand concepts of screw jack, friction, law of moments.
- Understand the Two-Stroke & Four-Stroke Diesel Engines and Petrol Engines.
- Get the knowledge of various Refrigeration and Air- Conditioning Systems.
- Know about the working of various turbines and pumps.

List of Experiments

1. To verify the law of Force Polygon
2. To verify the law of Moments using Parallel Force apparatus. (simply supported type)
3. To determine the co-efficient of friction between wood and various surface (like Leather, Wood, Aluminum) on an inclined plane.
4. To find the forces in the members of Jib Crane.
5. To determine the mechanical advantage, Velocity ratio and efficiency of a screw jack.
6. To determine the mechanical advantage, Velocity ratio and Mechanical efficiency of the Wheel and Axle

7. To verify the law of moments using Bell crank lever.
8. To calculate the Mechanical Advantage, Velocity Ratio and Efficiency of Single Start, Double Start and Triple Start Worm & Worm Wheel.
9. To Study Two-Stroke & Four-Stroke Diesel Engines.
10. To Study Two-Stroke & Four-Stroke Petrol Engines.
11. To Study the vapor compression Refrigeration System.

ETEL 171A	COMMUNICATION SKILLS LAB	C
		1

Communication Skills Lab Activity

Activity 1: Self- introduction: Informal introduction & formal introduction; A detailed write up on formal 'Self Introduction'; Formal Introduction of oneself in front of the group.

Activity 2: News Reading: Introduction to 'online News papers'; Browsing and selecting the preferred Newspaper; Browsing through the News Headlines; Selecting interested News items; Comprehending the content, writing down the essence and reading the News in front of the Group. Discuss 5 to 8 new words or terms, 4 to 5 important personalities of that day's news etc.

Activity 3: JAM: Introduction to 'Just A Minute speech' and the 'Extempore speech'; Preparation of speech on given topic(different topic for each student); delivery of the speech; Feedback(on content, time management, body language etc. highlighting the positive aspects first.)

Activity 4: News Discussions: Selecting News of the day, Discussing among the group, prepare the news content and prepare the group opinion about the issue and present it in front of the class by the group involving each member; select 5 new words & new usages from the selected news item

Activity 5: Conversation ability: Characteristics of effective conversation; Listening to a few sample conversations; preparing conversation based on the given situation; enacting the situation through effective delivery of the script; feedback & suggestions for improvement.

Activity 6: Role Play: Characteristics of Role Play; assigning roles; developing the content to deliver; enacting the role with effective delivery; feedback & suggestions for improvement.

Activity 7: Public Speaking: Characteristics of effective Public speaking; possible barriers; watching demo online; topic assignment, information gathering & recording; delivery in front of the class; feedback & suggestions for improvement. . (Different topic for each student)

Activity 8: Group Discussion: Importance and characteristics; Dos & Don'ts in GD; Demo display; assign topic for the group, Preparation & performance; feedback & suggestions for improvement.

Activity 9: Debate: Difference between Group Discussion & Debating; Watching demo of Debating; Topic for the group of 2 or 4; preparation and performance; feedback & suggestions for improvement.

Activity 10: .Interview: Importance & purpose of Job Interview; Interview etiquettes; Watch demo interview; Appear for formal mock interview; feedback & suggestions for improvement.

ETCS 153A	PROGRAMMING FOR PROBLEM SOLVING LAB	C
		1

Course Overview:

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.

Objectives and Expected Outcomes:

At the end of course the students should be able to:

1. To understand the various steps in program development
2. To learn the syntax and semantics of C programming language
3. To use the structural programming approach in solving the problem.

LIST OF EXPERIMENTS

Lab1: Familiarization with programming environment
Lab 2: Simple computational problems using arithmetic expressions
Lab 3: Problems involving if-then-else structures
Lab 4: Iterative problems e.g., sum of series
Lab 5: 1D Array manipulation
Lab 6: Matrix problems, String operations
Lab 7: Simple functions
Lab 8 and 9: Programming for solving Numerical methods problems
Lab 10: Recursive functions
Lab 11: Pointers and structures
Lab 12: File operations

ETCS565A	COMPUTER CENTRE PLANNING & ESTABLISHMENT LAB	C
		1

Course Overview:

Computing and programming is essential to leverage the technical skills of a student. These techniques equip the students with know-how of the latest technologies and reduce considerable time in solving problems. The course of Information Technology Fundamentals has become essentially the present age of computer technology and information, as the applications of information technology can be found in all aspects of our lives.

Objectives and Expected Outcomes:

The main objective is to introduce IT in a simple language to all undergraduate students, regardless of their specialization. It will help them to pursue specialized programs leading to technical and professional careers and certifications in the IT industry. The focus of the subject is on introducing skills relating to IT basics, computer applications, programming, interactive media, Internet basics, etc. At the end of this course, students should be able to

- Understand basic concepts and terminology of information technology.
- Have a basic understanding of personal computers and their operations.

LIST OF EXPERIMENTS

1. MS-Windows: Operating system-Definition & functions, basics of Windows. Basic components of windows, icons, types of icons, taskbar, activating windows, using desktop, title bar, running applications, exploring computer, managing files and folders, copying and moving files and folders. Control panel – display properties, adding and removing software and hardware, setting date and time, screensaver and appearance. Using windows accessories.

2. Documentation Using MS-Word - Introduction to Office Automation, Creating & Editing Document, Formatting Document, Auto-text, Autocorrect, Spelling and Grammar Tool, Document Dictionary, Page Formatting, Bookmark, Advance Features of MS-Word-Mail Merge, Macros, Tables, File Management, Printing, Styles, linking and embedding object, Template.

3. Electronic Spread Sheet using MS-Excel - Introduction to MS-Excel, Creating & Editing Worksheet, Formatting and Essential Operations, Formulas and Functions, Charts, Advance features of MS-Excel-Pivot table & Pivot Chart, Linking and Consolidation, Database Management using Excel-Sorting,

Filtering, Table, Validation, Goal Seek, Scenario.

4. Presentation using MS-PowerPoint: Presentations, Creating, Manipulating & Enhancing Slides, Organizational Charts, Excel Charts, Word Art, Layering art Objects, Animations and Sounds, Inserting Animated Pictures or Accessing through Object, Inserting Recorded Sound Effect or In-Built Sound Effect.

SEMESTER II

BSPH132A	ELECTRICITY AND MAGNETISM	C
		4

Course Overview:

The course gives an overview and understanding of basic physics, with moderate use of mathematical formalism.

Topics include concepts in electricity and magnetism together with their relationship to practical applications.

The major concepts covered are: - The abstraction from forces to fields using the examples of the electric and magnetic fields, with some applications vis-a-vis the connection between conservative forces and potential energy, how charges move through electric circuits, the close connection between electricity and magnetism, leading to the discovery of electromagnetic waves. - The integral form of Maxwell's Equations etc.

Objective and Expected Outcome:

The objective of the course is to develop a basic understanding of electric and magnetic fields in free space: definitions, units, laws and rules as well as interesting things connected with discoveries and discoverers of crucial phenomena.

After completion of this course, students will:

- Apply knowledge of electricity and magnetism to explain natural physical processes and related technological advances.
- Use an understanding of calculus along with physical principles to effectively solve problems encountered in everyday life, further study in science, and in the professional world.
- The student can solve problems with moderate mathematical complexity related to electric and magnetic force and field, electric charge, electric potential, current, voltage and resistance, capacitors, electromagnetic waves, reflection, refraction, interference and diffraction.
- Design experiments and acquire data in order to explore physical principles, effectively communicate results, and critically evaluate related scientific studies.
- Assess the contributions of physics to our evolving understanding of global change and sustainability while placing the development of physics in its historical and cultural context.

Electric Field: Electric Field & Lines. Electric Field E due to a Ring of Charge. Electric Flux. Gauss's law. Gauss's law in Differential form. Applications of Gauss's Law

Electric Potential: Line Integral of Electric Field. Electric Potential Difference and Electric Potential V (Line integral). Conservative Nature of Electrostatic Field. Relation between E and V. Electrostatic Potential Energy of a System of Charges. Potential and Electric Field of (1) a Dipole, (2) a Charged Wire and (3) a Charged Disc. Force and Torque on a Dipole. Conductors in an Electrostatic Field.

Dielectric Properties of Matter

Dielectrics: Electric Field in Matter. Dielectric Constant. Parallel Plate Capacitor with a Dielectric. Polarization, Polarization Charges and Polarization Vector. Electric Susceptibility. Gauss's law in Dielectrics. Displacement vector D. Relations between the three Electric Vectors. Capacitors filled with Dielectrics.

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.

Magnetic Field:

Magnetic Effect of Currents: Magnetic Field B. Magnetic Force between Current Elements and Definition of B. Magnetic Flux. Biot-Savart's Law : B due to (1) a Straight Current Carrying Conductor and (2) Current Loop. Current Loop as a Magnetic Dipole and its Dipole Moment (Analogy with Electric Dipole). Ampere's Circuital law (Integral and Differential Forms): B due to (1) a Solenoid and (2) a Toroid. Properties of B. Curl and Divergence of B. Vector Potential.

Magnetic Properties of Matter:

Magnetism of Matter: Gauss's law of magnetism (Integral and Differential Forms). Magnetization current. Relative Permeability of a Material. Magnetic Susceptibility. Magnetization Vector (M). Magnetic Intensity (H). Relation between B, M and H. Stored Magnetic Energy in Matter. Magnetic Circuit. Electromagnetic induction: Faraday's law (Differential and Integral forms). Lenz's Law, self-inductance and mutual inductance.

TEXT BOOK

1. K.K. Tiwari, Electricity and Magnetism(S.Chand).

REFERENCE BOOKS:

1. J.H.Fewkes & John Yarwood Electricity and Magnetism. Vol. I (Oxford Univ. Press).
2. Arthur F. Kip, Fundamentals of Electricity and Magnetism (McGraw-Hill).
3. Edward M. Purcell, Electricity and Magnetism (McGraw-Hill Education).
4. D.C. Tayal, Electricity and Magnetism (Himalaya Publishing House).
5. David J. Griffiths, Introduction to Electrodynamics, 3rd Edn, (Benjamin Cummings).

OPEN ELECTIVE - II	C
	6

There are three open electives offered by other departments / schools in second semester. The department will permit student to opt one open elective based on choice of student and consent of the course advisor.

S.No.	Course Code	Course Title
1	IIIT104	Understanding The Power of Data
2	IIIT102	Fundamentals of Innovation and Entrepreneurship
3	ETCE12	URBAN Engineering

ETEL236A	SOFT SKILLS AND PERSONALITY DEVELOPMENT	C
		3

Course Overview:

In the contemporary world, employability and worth of a person depends largely on the ability of a person to exhibit different skills. Despite having sound knowledge of one's specialized subject, one cannot compete with his peer groups unless one has the potential of performance. It contains four modules which aim at improvising the soft skill, career management, language competence and behavioral skills of the students. The students will be trained in employment related written correspondence which would improve their potential in the corporate and academic world.

Objectives and Expected Outcomes

The course aims to spread basic awareness about the significance of soft skills in professional and inter-personal communications and facilitate an all-round development of personality. The objective is to make students learn the effective business, social and professional communication strategies. Students will be benefitted with pleasant and appealing personality traits as self-confidence, positive attitude, emotional intelligence, social grace, flexibility, friendliness and effective communication skills.

This course is designed to help the learners build their communication and personality skills in fostering skills are crucial to increase employment opportunities and to compete successfully in the business environment.

UNIT I

Soft Skills: Inter & intra personal qualities; Leadership; Team spirit; Decision Making; Effective listening for effective communication; Speeches by Martin Luther King & Charlie Chaplin (Listen to the speech and its pronunciation, tone and intonation; content analysis to know the message and also discuss the personality of the speaker).

UNIT II

Career Management: Resume and C V writing, Job applications, Interviews, Group Discussion Techniques, Negotiation and Meeting Skills, Presentation skills, Speeches by John F. Kennedy & G. Vivekananda (Listen to the speech and its pronunciation, tone and intonation; content analysis to know the message and also discuss the personality of the speaker).

UNIT III

Written Communication: Employment related correspondence, Report Writing, Editing the given document; Phonetics: Importance, Speech articulation, Transcription, Speeches by APJ Abdul Kalam & Shashi Tharoor (Listen to the speech and its pronunciation, tone and intonation; content analysis to know the message and also

discuss the personality of the speaker).

UNIT IV

Etiquettes & Manners: Effective non-verbal communication; Hospitality Tips; Corporate behaviour; Etiquettes at Business Meetings; Transcultural Etiquettes; Speeches by Barack Obama & Narendra Modi (Listen to the speech and its pronunciation, tone and intonation; content analysis to know the message and also discuss the personality of the speaker).

TEXT BOOK:

Prasad, P. Functional Aspects of .Communication Skills.4th Ed. S.K. Kataria & Sons, New Delhi.2008

REFERENCE BOOKS:

1. Sinha, K.K. Business Communication. Galgotia Publishers.
2. Mitra, Barun K. Personality Development and Soft Skills. Oxford University Press.
3. Bansal, R.K. and Harrison J.B. Spoken English for India: A Manual of Speech and Phonetics, Hyderabad: Orient Longman.
4. “Best Poems”, http://100.best-poems.net/. 20 July 2016.
5.“Classic English Short Stories”, http://www.eastoftheweb.com/short-stories/Collections/ClasEngl.shtml, 20 July 2016

ETCH 125A	ENVIRONMENTAL STUDIES	C
		3

Course Overview:

Everything that surrounds and affects living organisms is environment. Environment includes all those things on which we are directly or indirectly dependent for our survival, whether it is living or biotic components like animals, plants or non-living or abiotic components like soil, air and water etc. It belongs to all, influences all and is important to all.

Environmental Protection Act (1986) defined "Environment as the sum total of water, air and land, their interrelationship among themselves and with the human beings, other living organisms and materials." Environmental studies are important since it deals with the most mundane problems of life like hygienic living conditions, safe and clean drinking water, fresh air, healthy food and sustainable development.

The syllabus for Environmental Studies includes conventional classroom teaching as well as field work. In this course the teacher simply acts as a catalyst to infer what the student observes or discovers in his/her own environment. Involvement of students in project work is one of the most effective learning tools for environmental issues. This syllabus is beyond the scope of textbook teaching and also the realm of real learning by observing the surroundings. The content of this course provides an overview of introduction to environment, concept of an ecosystem, various renewable and non-renewable resources, how various biodiversity occur and different means to conserve these. This course also includes various types of pollution and environmental policies & practices related with environs. Finally, it also highlights the relationship of human population with environment. The course further integrates to project work such as visit to an area to document environmental assets river/ forest/ grassland/ hill/ mountain, visit to a local polluted site-Urban/Rural/Industrial/Agricultural, study of common plants, insects, birds, and study of simple ecosystems. These studies are as imperative as it forms a unique synergistic tool for comprehensive learning process. This will help students to recognize and appreciate how the technological advancement at global level, exponential growth of human population and their unlimited demands has put the environment at stake and has contaminated the environment worldwide.

Objective and Expected Outcome:

The main objective of the course is to create consciousness among the students with the idea about healthy and safe environment. This course is aimed to explain students that the rapid industrialization, crazy consumerism and over-exploitation of natural resources have resulted in degradation of earth at all levels. These changes need the discussion, concern and recognition at national and international level with respect to formulate protection acts and sustainable developments policies. It can be possible only if every citizen of the nation is environmentally educated and gets involved into this matter at the grass root level to mitigate pollution.

After studying the course, the learners will be able to comprehend and become responsive regarding environmental issues. They 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. This is the only inheritance which every genera of specie passes to their future generation.

UNIT I

Introduction of Environmental Studies: Multidisciplinary nature of environmental studies; Scope and importance; Concept of sustainability and sustainable development.

Natural Resources: Renewable and Non-renewable Resources

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

Ecosystems: 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: Levels of 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

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.

Environmental Policies and practices: 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. International agreements: Montreal & Koyoto protocol and convention on biological diversity. Nature reserves, tribal population and rights, human wild life conflicts in Indian context.

UNIT IV

Human Communities and the Environment: 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).

Field work:

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. Erach Bharucha, Textbook of Environmental Studies, Universities Press (P) Ltd., Hyderabad, India.
2. Anubha Kaushik and C. P. Kaushik, Environmental Studies, New Age International Publishers (P) Ltd. New Delhi.

REFERENCE BOOKS:

1. A.K. De, Environmental Chemistry, New Age International Publishers (P) Ltd. New Delhi.
2. P. H. Raven, D. M. Hassenzahl & L. R. Berg, Environment, John Wiley & Sons, New Delhi.
3. J. S. Singh, S. P. Singh and S. R. Gupta, Ecology, Environmental Science and Conservation, S. Chand Publication, New Delhi.

ETCS 316A	WEB TECHNOLOGIES	C
		4

Course Overview:

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.

Once the student has gained a basic knowledge of HTML coding and CSS, the goal of this course is to create large-scale, interactive, professional Web sites that are in accordance with current standards. 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. Other Web design topics include information architecture, scalability, multimedia integration, browser compatibility, standardization, and maintenance.

Objectives and Expected Outcomes:

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

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

UNIT I

Overview of the Internet and World Wide Web and Search Engines, Common terminology: IP Addressing, URLs, Domain names. Website Creation and maintenance, Web Hosting and Publishing Concepts.

Static Webpages: 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.

UNIT II

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

XML: Introduction to XML-Mark up languages, Features of Markup 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.

UNIT IV

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.

Textbooks:

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

Reference Books:

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.

ETCS112A	OBJECT ORIENTED PROGRAMMING	C
		4

Course Overview:

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.

Objectives and Expected Outcomes:

At the end of the course, the students should be able to:

On successful completion of this course students will be able to:

1. Explain the steps in creating an executable program for a computer, including the intermediate representations and their purpose
2. Manipulate binary patterns and understand the use of binary to represent numbers.
3. Apply good programming style and understand the impact of style on developing and maintaining programs
4. Effectively use a version control system and the Linux command line tools for incremental development.
5. Explain the benefits of object-oriented design and understand when it is an appropriate methodology to use.
6. Design object-oriented solutions for small systems involving multiple objects.
7. Identify the relative merits of different algorithmic designs.

UNIT I

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

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

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

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:

1. 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:

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. Yashwant Kanethkar, "Object Oriented Programming using C++", BPB

ETCA 164A	WEB TECHNOLOGIES LAB	C 1

Course Overview:

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.

Objectives and Expected Outcomes:

The following experiments are expected to be performed in the lab.

1. Write HTML/Java scripts to display your CV in Web Browser.
2. Creation and annotation of static web pages using any HTML editor.
3. Write a program to use XML and JavaScript for creation of your homepage.
4. Write a program in XML for creation of DTD which specifies a particular set of rules.
5. Create a Stylesheet in CSS/XSL and display the document in Web Browser.
6. Create a Registration Form with Table.
7. CSS : Inline Style , Internal Style ,and External Style Sheets
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
        - 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

ETCS166A	OBJECT ORIENTED PROGRAMMING LAB	C 1

Course Overview:

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.

Objective and Expected Outcome:

Students who have already studied Structural programming like C, would understand how Object-oriented programming would help them in coding in a simpler and better way. For instance, a user of the program should only know what the input is and what is the output, he should not be concerned about the process. The programmer implements the concept through abstraction and information hiding, which are important features of object-oriented programming. The course would help students achieve the following goals:

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

LIST OF EXPERIENTS

Q1. 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 value for n and an int value for p, and returns the result 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.

Q2. 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

Q 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
Do another (Y/ N) ? N

Q4. 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

Q 5. Create two classes DM and DB which store the value of distances. DM stores distances in metres and centimeters and DB in feet and inches. Write a program that can read values for the class objects 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 metres and cenitmetres depending on
the object on display.

Q 6. Create a class rational which represents a numerical value by two double values- NUMERATOR & 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.

Q 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.

Q 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.

Q9. A hospital wants to create a database regarding its indoor patients. The information to store include

a) Name of the patient

b) Date of admission

c) Disease

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 (less than twelve years in age).

Q 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.

Q11. 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.

Q12. 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)".

Q13. 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.

Q 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

BSPH132A	ELECTRICITY AND MAGNETISM LAB	C
		2

Hands On Experiments based on the theory **BSPH132A**

SEMESTER III

ETEC210A	DIGITAL ELECTRONICS	C
		4

Course Overview

Lectures and 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.

Objective and Expected Outcome:

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

COURSE OUTCOME:

- Create the appropriate truth table from a description of a combinational logic function.
- Create a gate-level implementation of a combinational logic function described by a truth table using and/or/inv gates, muxes or ROMs, and analyze its timing behavior.
- 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.
- Describe the operation and timing constraints for latches and registers.
- 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).
- Evaluate combinational and sequential logic designs using various metrics: switching speed, throughput/latency, gate count and area, energy dissipation and power
- Properly incorporate synchronous and asynchronous memories into a circuit design.
- Discuss how to interface digital circuits with analog components (ADC, DAC, sensors, etc.).

UNIT – I

Number Systems and Codes: Review of number systems, BCD codes and arithmetic, Gray code, self-complimenting codes, Error detection and correction.

Digital Circuits: Switching algebra & simplification of Boolean expressions, De Morgan‘s Theorem, Implementation of Boolean expressions (using logic gates)

UNIT – II

Combinational Logic Design: Combinational circuit designing, Minimization Techniques of Boolean functions such as Karnaugh map and Quine-Mc Cluskey methods, Arithmetic circuits, code convertors, multiplexers, demultiplexers, encoders, decoders & comparators. Parity generators and checkers.

Introduction to Sequential Logic: Need for sequential circuits, Binary cell, Latches and different types of Flip-Flop and their Conversions.

UNIT – III

Synchronous Sequential Circuit : Fundamentals of Synchronous sequential circuits, Analysis of Synchronous Sequential circuits, Design of Synchronous and Asynchronous Counters, Shift registers & Ring counters, Timing issues in synchronous circuits.

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.

UNIT – IV

Synchronous Machines: Classification of synchronous machines, Analysis and design of Finite State Machines.

Logic Families: Performance metrics of logic gates, Basic Transistor-Transistor Logic and CMOS logic.

Converters: A/D and D/A converters and their types.

TEXT BOOKS:

1. G.K. Kharate -Digital Electronics, Oxford University Press
2. Aanand Kumar -Fundamentals of Digital Circuits, Prentice Hall of India

REFERENCE BOOKS:

1. Morris Mano, ―Digital design, Prentice Hall of India
2. R.P. Jain- Modern Digital Electronics, Tata Mc Graw Publishers

3. Floyd- Digital Fundamentals, Pearson Publication

BSMA217A	REAL ANALYSIS	C
		4

Course Overview:

Much of mathematics relies on our ability to be able to solve equations, if not in explicit exact forms, then at least in being able to establish the existence of solutions. To do this requires knowledge of so-called "analysis", which in many respects is just Calculus in very general settings. The foundations for this work are commenced in Real Analysis, a course that develops this basic material in a systematic and rigorous manner in the context of real-valued functions of a real variable. Topics covered are: Basic set theory. The real numbers and their basic properties. Sequences: convergence, subsequences, Cauchy sequences. Open, closed, and compact sets of real numbers. Continuous functions and uniform continuity. The Riemann integral. Differentiation and Mean Value theorems. The Fundamental Theorem of Calculus. Series. Power series and Taylor series. Convergence of sequences and series of functions.

Objective and Expected Outcome:

A lot of mathematics is about real-valued continuous or differentiable functions and this generally falls under the heading of "real-analysis". Everything from ODEs and PDEs, Taylor series, Fourier transforms, and other functional decompositions. In addition, real-analysis is necessary for probability theory, which is the foundation for all of statistics, operations research, queuing theory, and the mathematical finance (e.g. Black Scholes Theory).Real Analysis enables the necessary background for Measure Theory.Measure theory is further used in the study of Stochastic Differential Equations (Finance, Signal Processing), Stochastic Geometry (Wireless Communications), Topology (Topological Data Analysis) and many more.

UNIT I

Review of Algebraic and Order Properties of R, ε-neighborhood of a point in R, Idea of countable sets, uncountable sets and uncountability of R. Bounded above sets, Bounded below sets, Bounded Sets, Unbounded sets.

UNIT II

Suprema and Infima, The Completeness Property of R, The Archimedean Property, Density of Rational (and Irrational) numbers in R, Intervals. Limit points of a set, Isolated points, Illustrations of Bolzano-Weierstrass theorem for sets.

UNIT III

Sequences, Bounded sequence, Convergent sequence, Limit of a sequence. Limit Theorems, Monotone Sequences, Monotone Convergence Theorem. Subsequences, Divergence Criteria, Monotone Subsequence Theorem (statement only), Bolzano Weierstrass Theorem for Sequences. Cauchy sequence, Cauchy’s Convergence Criterion.

UNIT IV

Infinite series, convergence and divergence of infinite series, Cauchy Criterion, Tests for convergence: Comparison test, Limit Comparison test, Ratio Test, Cauchy's nth root test, Integral test, Alternating series, Leibniz test, Absolute and Conditional convergence.

TEXT BOOK:

S.C. Malik and Savita Arora*; Mathematical Analysis;* New Age Science.

REFERENCE BOOKS:

1. R.G. Bartle and D. R. Sherbert; *Introduction to Real Analysis*; John Wiley and Sons Pvt. Ltd.
2. Gerald G. Bilodeau, Paul R. Thie, G.E. Keough; *An Introduction to Analysis*; Jones& Bartlett.
3. Brian S. Thomson, Andrew. M. Bruckner and Judith B. Bruckner; *Elementary Real Analysis,* Prentice Hall.

ETMA208A	NUMERICAL ANALYSIS	C
		4

Course Overview:

The purpose of this course is to sharpen the skills of students to solve problems related to engineering, social and medical science by using computer programming. We build mathematical models to describe real-life problems but the methods we studied in calculus, algebra and other areas of mathematics not necessarily provide solutions to these problems. For instance, if we have an equation involving transcendental functions such as e^x, $sin\ sin\ x$ then the methods we studied so far fail to find a root of this equation. In order to overcome this difficulty, we start with an initial estimate and improve this estimated value in an iterative manner. This is an example of how we solve a problem numerically.

Mathematical models and their numerical solutions are crucial in many problems of our life. This course introduces various methods to numerically solve the problems in different areas. In addition, the students will learn how effective a method is in solving a problem at hand.

Objective and Expected Outcome:

Differential and integral calculus is introduced to the students at intermediate level and they have an instinct of its importance in our life. Finding roots of equations arise in problems in optimization and problems in the stock market require least-squares solutions. Models in economics require a detailed analysis of system linear of equations. These problems are mathematically complex and require plenty of computational skills. During the course, students will learn numerical techniques to solve problems in differentiation, integration and differential equations. Further, students would be able to solve linear and non-linear equations iteratively.

This course aims to develop computational thinking in students and enable them to examine the algorithmic techniques to learn how computational tools can find the answer. In this way, students would even have a better understanding of some theoretical concepts in mathematics such as limits, sequence and their convergence. They will also become more efficient in detecting and controlling the errors, and can determine the accuracy of the results. Students are free to choose their own programming language so that the programming language is not a barrier.

After completing the course, students will have a good bird's-eye view of numerical analysis, and consequently, they will be better prepared for a more intense study of the subject.

UNIT I

Algorithms, Convergence, Errors: Relative, Absolute, Round off, Truncation.

Transcendental and Polynomial equations: Bisection method, Newton's method, Secant method.Rate of convergence of these methods.

UNIT II

System of linear algebraic equations: Gaussian Elimination and Gauss Jordan methods. Gauss Jacobi method, Gauss Seidel method and their convergence analysis.

UNIT III

Interpolation: Lagrange and Newton's methods. Error bounds. Finite difference operators. Gregory forward and backward difference interpolation.

UNIT IV

Numerical Integration: Trapezoidal rule, Simpson's rule, Simpsons 3/8th rule, Boole's Rule, Midpoint rule, Composite Trapezoidal rule, Composite Simpson's rule.

Ordinary Differential Equations: Euler's method, Runge-Kutta methods of orders two and four.

TEXT BOOK:

B. S. Grewal, *Numerical Methods in Engineering and Science*, Khanna Publishers.

REFERENCE BOOKS:

1. M. K. Jain, S.R.K. Iyengar and R.K. Jain; *Numerical Methods for Scientific and Engineering Computation*; New age International Publisher.
2. Brian Bradie; *A Friendly Introduction to Numerical Analysis*; Pearson Education.
3. C.F. Gerald and P.O. Wheatley; *Applied Numerical Analysis*; Pearson Education.
4. Uri M. Ascher and Chen Greif; *A First Course in Numerical Methods*; PHI Learning Private Limited.
5. John H. Mathews and Kurtis D. Fink; *Numerical Methods using Matlab;* PHI Learning Private Limited.

ETCS219A	FOUNDATION OF COMPUTER SYSTEMS	C
		4

Course Overview:

Throughout the course, students will be expected to demonstrate their understanding of Discrete Mathematics by being able to do each of the following:

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.

Objectives and Expected Outcomes:

1. For a given logic sentence express it in terms of predicates, quantifiers, and logical connectives
2. For a given a problem, derive the solution using deductive logic and prove the solution based on logical inference
3. For a given a mathematical problem, classify its algebraic structure
4. Evaluate Boolean functions and simplify expressions using the properties of Boolean algebra
5. Develop the given problem as graph networks and solve with techniques of graph theory.

UNIT I

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

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

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

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.

REFERENCES BOOKS:

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.

ETCS217A	DATA STRUCTURES	C
		4

Course Overview:

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.

Objectives and Expected Outcomes:

1. For a given algorithm student will able to analyze the algorithms to determine the time and computation complexity and justify the correctness.
2. For a given Search problem (Linear Search and Binary Search) student will able to implement it.
3. For a given problem of Stacks, Queues and linked list student will able to implement it and analyze the same to determine the time and computation complexity.
4. Student will able to write an algorithm Selection Sort, Bubble Sort, Insertion Sort, Quick Sort, Merge Sort, Heap Sort and compare their performance in term of Space and Time complexity.
5. Student will able to implement Graph search and traversal algorithms and determine the time and computation complexity.

UNIT I

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

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

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

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 Algorithms)

TEXT BOOKS:

1. E. Horowitz and S. Sahani, “Fundamentals of Data Structures”, GalgotiaBooksource Pvt. Ltd.
2. R. L. Kruse, B. P. Leung, C. L. Tondo, “Data Structures and program design in C”, PHI.

REFERENCES BOOKS:

1. Schaum’s outline series, “Data Structure”, TMH.
2. Y. Langsamet. al., “Data Structures using C and C++”, PHI.
3. Yashwant Kanetkar, “Data Structure through C”, BPB.

ETCS 257A	DATA STRUCTURES LAB	C
		1

Based on theory subject **ETEC217A** following experiments are to be performed:

LIST OF EXPERIMENTS

1. Write a program for multiplication and transpose of array.
2. Write a program to compute the transpose of a sparse matrix
3. Write a program to implement push and pop operation in Stack.
4. Write a program to convert a Infix notation to post fix notation using stacks
5. Write a program to evaluate postfix notation using stacks
6. Write a program to implement a linear queue
7. Write a program for swapping two numbers using call by value and call by reference strategies.
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.
9. Write a program to implement a linear search arrays and linked list.
10. Using iteration and recursion concepts write programs for finding the element in the array using the Binary search method.
11. Write the programs to implement bubble sort.
12. Write a program using iteration and recursion concepts for quick sort.
13. Write a program to implement merge sort.
14. Write a program to simulate various tree traversal techniques.
15. Write a program to simulate various BFS and DFS.

ETEC256A	DIGITAL ELECTRONICS LAB	C
		1

Hands-on experiments related to the course contents ETEC210A by performing experiments as given below:

- 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

BSMA351A	NUMERICAL ANALYSIS LAB	C
		1

Hands-on experiments related to the course contents **ETMA208A** by performing experiments as given below:

PRACTICAL/LAB WORK

(Modeling of the following problems using Matlab/ Mathematica/ Maple Etc.)

List of Practicals

(i) Calculate the sum 1/1 + 1/2 + 1/3 + 1/4 + ----------+ 1/ N.

(ii) To find the absolute value of an integer.

(iii) Enter 100 integers into an array and sort them in an ascending order.

(iv) Bisection Method.

(v) Newton Raphson Method.

(vi) Secant Method.

(vii) Regulai Falsi Method.

(viii) LU decomposition Method.

(ix) Gauss-Jacobi Method.

(x) SOR Method or Gauss-Siedel Method.

(xi) Lagrange Interpolation or Newton Interpolation.

(xii) Simpson's rule.

Note: For any of the CAS (Computer aided software) Data types-simple data types, floating data types, character data types, arithmetic operators and operator precedence, variables and constant declarations, expressions, input/output, relational operators, logical operators and logical expressions, control statements and loop statements, Arrays should be introduced to the students.

SEMESTER IV

BSPH236 A	ANALOG SYSTEMS AND APPLICATIONS	C 4

Overview:

The analog nature of electronic signals is of importance as the real world is analog, Our world is analog and requires us to deal with ever varying physical phenomena (light, sound, touch, taste etc.) Analog systems have many applications in real world such as amplifiers, sensors, voltage regulators, power supplies, oscillators, image processing etc. This course deals with basics of analog components such as semiconductor diodes, transistors, power amplifiers, operational amplifiers etc. and their various applications.

Objective and Expected Outcome:

Students will learn basic concepts of junction diode, their types and various applications. They should be able to demonstrate use of pn junction diode as wave shaping/clipping circuits in computers, radios, radars etc., as switches in digital logic designs, as clamping circuits in TV receivers as well as voltage multipliers, and as rectifiers in DC power supply manufacturing. They should understand the working of Zener diode and its application in voltage regulation, LED, photodiode and solar cell. After successful completion of this course, students will be able to understand basic design of amplifiers and oscillators using bipolar junction transistor and feedback. They will be able to design simple circuits such as adder, subtractor, differentiator, integrator, Log Amplifier, Zero Crossing detector, Wien Bridge Oscillator using Operational Amplifier.

Syllabus:

Junction diode, their types and applications. Use of pn junction diode as wave shaping/clipping circuits , clamping circuits and as rectifiers. Working of Zener diode and its application in voltage regulation, LED, photodiode and solar cell.

BJT transistor working, biasing circuits in BJT, Basic design of amplifiers and oscillators using bipolar junction transistor and feedback, Class A, B, C and AB amplifiers.

Operational Amplifiers, open loop and closed loop configuration, feedback in amplifiers, types of feedback circuits, barkhausen criterion for sustained oscillations.

Amplifiers, Adder, subtractor, differentiator, integrator, Log Amplifier, Zero Crossing detector, Wien Bridge Oscillator using Operational Amplifier.

BSMA137A	ALGEBRA	C
		4

Course Overview:

Algebra is designed to give students a foundation for all future mathematics courses. The fundamentals of algebraic problem-solving are explained. Students will explore the basic concepts of matrices, relation between the roots and coefficients of general polynomial equation in one variable, Nature of roots by inspection of change of sign of equations. Throughout the course the student learns how to apply the concepts in real-life situations.

Objective and Expected Outcome:

During this course the student will be able to find the rank, Eigen values of matrices and solve the homogeneous and non-homogeneous systems, solution of cubic and biquadratic equations.

Upon completion of this course, you should be able to students understand mathematical concepts, symbols and procedures and are able to apply them to real world situations. Use appropriate mathematical concepts and skills to solve problems in both familiar and unfamiliar situations including those in real-life contexts. Select and apply appropriate inquiry and mathematical problem solving techniques in recognize patterns.

The main objective of this program is to cultivate a mathematical aptitude and nurture the interests of the students towards problem solving aptitude. Further, it aims at motivating the young minds for research in mathematical sciences and to train computational scientists who can work on real life challenging problems. It further aims at motivating the students to join teaching profession.

UNIT I

Symmetric; Skew-symmetric; Hermitian and skew Hermitian matrices; Elementary Operations on matrices; Rank of a matrices; Inverse of a matrix; Linear dependence and independence of rows and columns of matrices; Row rank and column rank of a matrix; Eigenvalues, eigenvectors and the characteristic equation of a matrix; Minimal polynomial of a matrix; Cayley Hamilton theorem and its use in finding the inverse of a matrix.

UNIT II

Applications of matrices to a system of linear (both homogeneous and non–homogeneous) equations; Theorems on consistency of a system of linear equations; unitary and Orthogonal Matrices; Bilinear and Quadratic forms

UNIT III

Relations between the roots and coefficients of general polynomial equation in one variable; Solutions of polynomial equations having conditions on roots; Common roots and multiple roots; Transformation of equations

UNIT IV

Nature of the roots of an equation; Descarte’s rule of signs; Solutions of cubic equations (Cardon’s method);

Biquadratic equations and their solutions.

TEXT BOOK:

Chandrika Prasad; *Text Book on Algebra and Theory of Equations*; Pothishala Private Ltd., Allahabad

REFERENCE BOOKS:

1. David C. Lay; *Linear Algebra and its Applications*; Pearson Education.
2. H.S. Hall and S.R. Knight; *Higher Algebra, H.M. Publications*; New Delhi.
3. Shanti Narayan; *A Text Books of Matrices*; S.Chand, New Delhi.

ETCS 220A	ANALYSIS AND DESIGN OF ALGORITHM	C
		4

Course Overview:

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

Objectives and Expected Outcome:

At the end of the course students the student should be able to

1. Analyze the asymptotic performance of algorithms.
2. Write rigorous correctness proofs for algorithms.
3. Demonstrate a familiarity with major algorithms and data structures.
4. Apply important algorithmic design paradigms and methods of analysis.
5. Synthesize efficient algorithms in common engineering design situations.

UNIT I

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

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

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

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.

ETMC226A	FUNDAMENTALS OF MANAGEMENT	C
		3

Course Overview:

Technical skills alone do not meet the real-world work and the business requirements; they have to be supplemented by management training. In fact, most of the people find that their success depends as much on general management skills and understanding operational systems as on their technical expertise. To become complete professional, students need a firm foundation in these basic managerial skills.

Fundamentals of Management are a basic introductory and foundational management course for under graduates. This course is designed for students to equip themselves with key knowledge, skills and competencies in various aspects of management. This course enables the students to develop an understanding of management and organization and focuses on important management functions such as planning, organizing, leading and controlling for successful managerial activities. The students will learn how successful managers use organizational resources through organizational functions in order to effectively and efficiently achieve organizational objectives.

Specific techniques related to managerial functions are explored as well as the broad issues and trends influence the practice of contemporary management, globalization, technology, diversity, and competitive advantage. Special emphasis is on basics of all the departments in the organization like Human Resource Management, Marketing Management, Productions and Operations Management and Financial Management.

Objectives and Expected Outcomes:

The objective of this course is for each student to be able to know, comprehend, apply, analyze, synthesize and evaluate the basic fundamentals of managing organizations. Through the learning of this course on fundamentals of management, students will gain fundamental knowledge and skills for management in contemporary organizations. These include the “How to” and “Why”. Students will also develop analytical and critical thinking skills in the context of contemporary organizations. This focuses on the entire organisation from both a short term and long-term perspective for strategic vision, objectives, crafting a strategy and implementing it. Specifically, the learning objectives for the students are:

- Demonstrate basic knowledge of management and organization.
- Demonstrate a basic understanding of management functions such as planning, organizing, leading and controlling; and how successful managers effectively and efficiently use these functions and their business resources to achieve organizational objectives.
- Develop knowledge of fundamental management concepts and skills.
- Identify the key competencies needed to be an effective manager.
- Identify the most important components of human resource planning; outline a model of organizational staffing; recruitment; selection; orientation; human resource planning and training.

ETCS307A	DATABASE MANAGEMENT SYSTEMS	C
		4

Course Overview:

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.

Objectives and Expected Outcomes:

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 expressions.
7. For a given specification of the requirement design the databases using E-R method and normalization.
8. For a given specification construct the SQL queries for Open source and Commercial DBMS - MYSQL, ORACLE, and DB2.
9. For a given query optimize its execution using Query optimization algorithms
10. For a given transaction-processing system, determine the transaction atomicity, consistency, isolation, and durability.
11. Implement the isolation property, including locking, time stamping based on concurrency control and Serializability of scheduling.

UNIT I

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

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

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

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.

SUGGESTED 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.
3. “Fundamentals of Database Systems”, R. Elmasri and S. Navathe, Pearson Education

BSPH236A	ANALOG SYSTEMS AND APPLICATIONS LAB	C
		2

Hands-on experiments related to the course contents BSPH236A

ETCS 355A	DATABASE MANAGEMENT SYSTEMS LAB	C
		1

Course Overview:

A database management system (DBMS) is computer application software that provides a way to manage data. The requirement of modern days is to have an automated system that manages, modifies and updates data accurately. This is achieved by a DBMS in robust, correct and non-redundant way. DBMS lab aims at practicing and achieving this aim by using various software's such as SQL, ORACLE, and MS – Access etc. All these require a thorough practice of various DDl, DCL and DML queries.

LIST OF EXPERIMENTS

1. Design a Database and create required tables. For e.g. Bank, College Database
2. Apply the constraints like Primary Key, Foreign key, NOT NULL to the tables.
3. Write a SQL statement for implementing ALTER, UPDATE and DELETE
4. Write the queries to implement the joins
5. Write the queries for implementing the following functions: MAX (), MIN (), AVG (), COUNT ()
6. Write the queries to implement the concept of Integrity constrains
7. Write the queries to create the views
8. Perform the queries for triggers
9. Perform the following operation for demonstrating the insertion, updation and deletion using the referential integrity constraints

ETCA365A	LINUX ENVIRONMENT LAB	C
		1

Course Overview:

This course will prepare students to develop software in and for Linux/UNIX environments. Topics to be covered include basic operating system concepts, effective command line usage, shell programming, the C language, programming development tools, system programming, network programming (client-server model and sockets), and GUI programming

Objectives and Expected Outcomes:

At the end of the students should be able to:

1. Understanding the basic set of commands and utilities in Linux/UNIX systems.
2. To learn to develop software for Linux/UNIX systems.
3. To learn the C language and get experience programming in C.
4. To learn the important Linux/UNIX library functions and system calls. •To understand the inner workings of UNIX-like operating systems. •To obtain a foundation for an advanced course in operating systems

LIST OF EXPERIMENTS

1. **Installing Linux :**Installing the Operating System
2. **Exploring the System :**Starting Up and changing runlevels, Using the man utility, Using built-in help switches for commands, Using Auto completion
3. **Common System Utilities:** Using cd**,** Using pwd**,** Using mkdir**,** Using rmdir, Using Touch, Using ls**,**Using mv**,**Using cp**,** Using cat, Using Redirection**,**rm**,**Using vi**,**Searching for files: grep, frep and similar commands
4. **The XWindow System:** Preamble, Virtual terminals**,** Setting up a basic display**,**X clients**,** Window Managers**,** Display Manager**,** xinit and startx**,** system-config-display
5. **The Shell and Shell Scripting :** Different kind of shells (c shell, bash shell, korn shell and others), A simple Script, Using variables in scripts, Using Control Structures
6. **User Accounts :** PreambleManually 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.
7. **Managing and Installing Software :**Installing, Querying and Uninstalling PackagesThird party tools,Building Software from Source

8. **Understanding Devices:** Determining device type, Creating devices, mounting and umounting devices

SEMESTER V

ETCS304A	COMPUTER NETWORKS	C
		4

Course Overview:

The subject of computer networking is enormously complex, involving many concepts, protocols, and technologies. To cope with the scope and complexity these protocols and technologies are woven together in an intricate manner in what is called the layered protocol stack (or suite). The layered organization allows breaking down complex functions required for computers networking into manageable tasks. This course is an introduction to computer networking using a top-down approach—that is, by beginning at the highest layer of the protocol stack (application layer) and proceeding down through different layers towards the lowest one (the physical layer). The course places emphasis on the application layer (a "high growth area" in networking). The course uses the Internet's architecture and protocols as the primary vehicle for studying fundamental computer networking concepts. More than often, the course will also include concepts and protocols from other network architectures. But the main focus is on the Internet, a fact reflected in organizing the course around the Internet's five-layer architecture.

Objectives and Expected Outcomes:

1. To develop an understanding of modern network architectures from a design and performance perspective.
2. To introduce the student to the major concepts involved in wide-area networks (WANs), local area networks (LANs) and Wireless LANs (WLANs).
3. To provide an opportunity to do network programming
4. Explain the functions of the different layer of the OSI Protocol.
5. 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

UNIT I

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

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

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

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.

ETCS211A	OPERATING SYSTEMS	C
		4

Course Overview:

To learn the fundamentals of Operating Systems.

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

Objectives and Expected Outcomes:

1. Create processes and threads.
2. Develop algorithms for process scheduling for a given specification of CPU utilization, Throughput, Turnaround Time, Waiting Time, Response Time.
3. For a given specification of memory organization develop the techniques for optimally allocating memory to processes by increasing memory utilization and for improving the access time.
4. Design and implement file management system.
5. For a given I/O devices and OS (specify) develop the I/O management functions in OS as part of a uniform device abstraction by performing operations for synchronization between CPU and I/O controllers.

UNIT I

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

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

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

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

REFERENCES BOOKS:

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

ETCS 206A	COMPUTER GRAPHICS	C
		4

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.

Objectives and Expected Outcomes: At the end of the course students the student should be able to

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

UNIT I

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

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

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

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.

BSMA326A	OPERATIONAL RESEARCH	C
		4

Course Overview:

Operations research is a mathematical science concerned with optimal decision making and the modeling of deterministic and probabilistic systems. Its focus and field of application are interdisciplinary, embracing a broad range of quantitative techniques. Industrial engineering is concerned with the design, improvement, and installation of integrated systems of personnel, material, and equipment. Together, operations research and industrial engineering provide a rational approach to engineering and managerial problem solving through deliberate application of scientific methods. Operations research helps in solving problems in different environments that needs decisions. The module covers topics that include: linear programming, Transportation, Assignment, and CPM/ MSPT techniques. Analytic techniques and computer packages will be used to solve problems facing business managers in decision environments. Course Objectives: This module aims to introduce students to use quantitate methods and techniques for effective decisions–making; model formulation and applications that are used in solving business decision problems.

Objective and Expected Outcome:

The purpose of the course is to provide students with the concepts and tools to help them understand the operations research and mathematical modeling methods. These methods will help the students to solve economic issues, which help to make a decision. The main goal is to find the lowest cost or the greatest profit in many linear programming in the economic field issues. The model was used to resolve the issue of transport. There are many of the problems regarding the transfer of goods within a minimum of expenses or the distribution of goods to obtain the maximum profit. It was used as a matter of allocation of activating the role of the distribution functions optimally to get the desired goal as costs or profits

Introduction to Operations Research (OR) • Introduction to Foundation mathematics and statistics • Linear Programming (LP), LP and allocation of resources, LP definition, Linearity requirement • Maximization Then Minimization problems. • Graphical LP Minimization solution, Introduction, Simplex method definition, formulating the Simplex model. • Linear Programming – Simplex Method for Maximizing. • Simplex maximizing example for similar limitations, Mixed limitations • Example containing mixed constraints, Minimization example for similar limitations. • Sensitivity Analysis: Changes in Objective Function, Changes in RHS, The Transportation Model •

Basic Assumptions. • Solution Methods: Feasible Solution: The Northwest Method, The Lowest Cost Method; Optimal Solution: The Stepping Stone Method, Modified; Distribution (MODI) Method. • The Assignment Model:- Basic Assumptions • Solution Methods:-Different Combinations Method, Short-Cut Method (Hungarian Method)

Knowledge and understanding - Be able to understand the characteristics of different types of decision-making environments and the appropriate decision making approaches and tools to be used in each type.

The program in operations research is designed to allow students to develop the technical, analytic, and managerial skills necessary to perform these tasks successfully. The graduate program in Operations Research & Industrial Engineering offers M.Sc. and Ph.D. degrees.

On completion of this course you should be able to:

1. Define and formulate linear programming problems and appreciate their limitations.
2. Solve linear programming problems using appropriate techniques and optimization solvers, interpret the results obtained and translate solutions into directives for action.
3. Conduct and interpret post-optimal and sensitivity analysis and explain the primal-dual relationship.
4. Develop mathematical skills to analyses and solve integer programming and network models arising from a wide range of applications.
5. Effectively communicate ideas, explain procedures and interpret results and solutions in written and electronic forms to different audiences.

UNIT I

Definition, scope, methodology and applications of OR. Types of OR models. Concept of optimization, Linear Programming: Introduction, Formulation of a Linear Programming Problem (LPP), Requirements for an LPP, Advantages and limitations of LP. Graphical solution: Multiple, unbounded and infeasible solutions.

UNIT II

Principle of simplex method: standard form, basic solution, basic feasible solution. Computational Aspect of Simplex Method: Cases of unique feasible solution, no feasible solution, multiple solutions and unbounded solution and degeneracy. Two Phase and Big-M methods.

UNIT III

Duality in LPP, primal-dual relationship. Transportation Problem: Methods for finding basic feasible solution of a transportation problem, Modified distribution method for finding the optimum solution, Unbalanced and degenerate transportation problems, transshipment problem, maximization in a transportation problem.

UNIT IV

Assignment Problem: Solution by Hungarian method, Unbalanced assignment problem, maximization in an assignment problem, Crew assignment and Travelling salesman problem.

Game Theory: Two person zero sum game, Game with saddle points, the rule of dominance; Algebraic, graphical and linear programming methods for solving mixed strategy games.

TEXT BOOKS:

1. Kanti Swarup, P.K. Gupta and Manmohan; *Operations Research*; Sultan Chand & Sons.
2. H.A.Taha*; Operations Research – An Introduction;Wiley.*

REFERENCE BOOKS:

1. Gupta, P.K. and Hira, D.S.; *Operations Research*; S. Chand & Co.
2. S.I. Gass; *Linear Programming* (3rd Edition); McGraw Hill, NY.
3. G. Hadley*; Linear Programming*; Narosa Publishing.

ETCS365A	COMPUTER NETWORKS LAB	C
		1

Course Overview:

Learn basic concepts of computer networking and acquire practical notions of protocols with the emphasis on TCP/IP. A lab provides a practical approach to Ethernet/Internet networking: networks are assembled, and experiments are made to understand the layered architecture and how do some important protocols work.

Objectives and Expected Outcomes:

After completing the course, the student should be able to:

1. Understand the structure and organization of computer networks; including the division into network layers, role of each layer, and relationships between the layers.
2. Execute and evaluate network administration commands and demonstrate their use in different network scenarios
3. Demonstrate the installation and configuration of network simulator.
4. Demonstrate and measure different network scenarios and their performance behavior.
5. Design and setup a organization network using packet tracer.

LIST OF EXPERIMENTS

1. Study of Network devices in detail
2. Connect the computers in Local Area Network using packet tracer
3. Implementation of Data Link Framing method - Character Count.
4. Implementation of Data link framing method - Bit stuffing and Destuffing.
5. Implementation of Error detection method - even and odd parity.
6. Implementation of Error detection method - CRC Polynomials.
7. Implementation of Data Link protocols - Unrestricted simplex protocol
8. Implementation of data link protocols - Stop and Wait protocol
9. Implementation of routing algorithms - Dijkstra’s algorithm
10. Study of Network IP Addressing using packet tracer
11. Design TCP client and server application to transfer file
12. Design UDP client and server application to transfer file
13. Working on Network Protocol Analyzer Tool (Ethereal/Wireshark)
14. Working on NMAP Tool for Port scanning.

ETCS 255A	OPERATING SYSTEMS LAB	C
		1

Hands-on experiments related to the course contents ETCS211A by performing All laboratory assignments

LIST OF EXPERIMENTS

Write down and execute the following programs using C/C++

1. Write programs using the following system calls of UNIX operating system fork, exec, getpid, exit, wait, close, stat, opendir, readdir.
2. Write C programs to simulate UNIX commands like ls, grep, etc.
3. Given the list of processes, their CPU burst times and arrival times, display/print the Gantt chart for FCFS and SJF. For each of the scheduling policies, compute and print the average waiting time and **average** turnaround time. (2 sessions)
4. 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. (2 sessions)
5. Developing Application using Inter Process Communication (using shared memory, pipes or message queues)
6. Implement the Producer –Consumer problem using semaphores (using UNIX system calls).
7. Implement some memory management schemes –I.
8. Implement any file allocation technique (Linked, Indexed or Contiguous).
9. Write programs based on shell programming

ETCS258A	COMPUTER GRAPHICS LAB	C
		1

Hands-on experiments related to the course contents ETCS 206A by performing All laboratory assignments

LIST OF EXPERIMENTS

1. Write a program to draw a point on screen. Study of various built in commands to draw basic objects on screen.
2. Write a program to implement Bresenham's Line Drawing Algorithm.
3. Write a program to implement various 2D, 3D Transformation matrices such as Translation, Scaling, Rotation, and Shear.
4. Write a program to implement Sutherland-Cohen line Clipping Algorithm.
5. Write a program to implement Bresenham's Circle Drawing Algorithm.
6. Write a program to implement Bezier Curves.
7. Write a program to implement B-Spline Curves.
8. Write a program to implement various Projections of 2D objects.
9. Write a program to implement various Projections of 3D objects.
10. Write a program to implement Isometric Projection.

ETCS507A	SEMINAR	C
		1

Students are expected to conduct a self-study under the supervision of the respective teachers. The study may be related to the subjects or of related area and be useful for the students. Students will be expected to give presentations regarding the same.

SEMESTER VI

ETCS506A	PYTHON PROGRAMMING	C
		4

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.

Objectives and Expected Outcomes:

The learning objectives of this course are:

- To understand why Python is a useful scripting language for developers.
- To learn how to design and program Python applications.
- To learn how to use lists, tuples, and dictionaries in Python programs.
- To learn how to identify Python object types.
- To learn how to use indexing and slicing to access data in Python programs.
- To define the structure and components of a Python program.
- To learn how to write loops and decision statements in Python.
- To learn how to write functions and pass arguments in Python.
- To learn how to build and package Python modules for reusability.
- To learn how to read and write files in Python.
- To learn how to design object-oriented programs with Python classes.
- To learn how to use class inheritance in Python for reusability.
- To learn how to use exception handling in Python applications for error handling.

Unit I

Introduction to Python: The basic elements of python, Branching Programs, Control Structures, Strings and Input, Iteration

Unit II

Functions, Scoping and Abstraction: Functions and scoping, Specifications, Recursion, Global variables, Modules, Files

Unit III

Classes and Object: Oriented Programming: Abstract Data Types and Classes, Inheritance, Encapsulation and Information Hiding, Handling Exceptions

Unit IV

Simple Algorithms and Data structures: Search Algorithms, Sorting
Algorithms, Hash Tables

Text Books:

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

Reference Books

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

ETCS202A	SOFTWARE ENGINEERING	C
		4

Course Overview:

Software engineering is the branch of computer science that creates practical, cost-effective solutions to computing and information processing problems, preferentially by applying scientific knowledge, developing software systems in the service of mankind. 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. The course will combine a strong technical focus with a capstone project providing the opportunity to practice engineering knowledge, skills, and practices in a realistic development setting with a real client. The program's goal is to provide a professionally guided education in software engineering that prepares graduates to transition into a broad range of career options: industry, government, computing graduate program, and professional education.

Objectives and Expected Outcomes:

All Software Engineering students will have demonstrated:

1. How to apply the software engineering lifecycle by demonstrating competence in communication, planning, analysis, design, construction, and deployment
2. An ability to work in one or more significant application domains
3. Work as an individual and as part of a multidisciplinary team to develop and deliver quality software
4. Demonstrate an understanding of and apply current theories, models and techniques that provide a basis for the software lifecycle.

UNIT I

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, Requirements documentation, Nature of SRS, Characteristics & organization of SRS.

UNIT II

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

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

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:

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.

ETCA324A	.NET FRAMEWORK	C
		4

Course Overview:

Microsoft Asp.Net Framework is a widely used development framework for building enterprise level web applications, that today's developers love to use. The Dot Net technology offers immeasurable benefits for various issues like memory management, security, and exceptional handling. The .NET Framework is a technology that supports building and running the next generation of apps and XML Web services. NET Framework. You can think of the runtime as an agent that manages code at execution time ,providing core services such as memory management, thread management, and remoting, while also enforcing strict type safety and other forms of code accuracy that ensure security and robustness. In fact, the concept of code management is a fundamental principle of the runtime Code that targets the runtime is known as managed code, while code that does not target the runtime is known as unmanaged code.

Objective and Expected Outcome:

The .NET Framework is designed to fulfill the following 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.

UNIT I

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

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

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

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:

1.ASP.NET, WroxPublications
2.ASP.NET and VB.NET, Wrox Publication
3.ASP.NET and C#.NET, Wrox Publication.

ETCS505A	TECHNICAL WRITING	C
		3

Course Overview:

This course will explore the ways that writing supports work in scientific and technical fields. Through a variety of assignments incorporating both written and visual formats, students in this course will learn effective strategies for responding to communication challenges, with special emphasis on audience analysis, document design, communication ethics, collaboration, professional style, and editing. You can expect to develop your ability to organize and craft information for manuals, journal articles, and reports and to learn about document design, production principles, interactive documentation, and desktop publishing. Technical writers are crucial for fields in engineering, software, and the sciences. Successful technical writers are good at translating science and technology for various audiences, including non-experts such as the end users of a product or customers who are willing to pay for technical solutions for a problem. Experts who work in technical and scientific fields can also benefit from familiarity and practice in technical writing.

Objectives and Expected Outcomes

In this course, students will become acquainted with the forms, functions, and rhetorical lives of technical documents. Students will examine and produce a variety of technical documents:

- Information reports and memos
- Usability reports
- Technical instructions
- Technical descriptions
- Technical reports
- Technical manuals

In the course of preparing these documents, candidates will consider concepts such as how audiences affect these texts, document design principles, and the role of visuals or graphics in both written and oral texts. Students will combine analysis, organization, and visual elements to pursue high standards in document design culminating in a major technical writing project. The main outcomes of the course are

1. To understand the genre and manipulate the structure of selected technical documents;
2. To convey clearly, cogently and correctly, through written media, the technical aspects of a practice to non-specialist audiences;
3. To recognize and use the rhetorical and stylistic elements necessary for the successful practice of scientific and technical communication;
4. To work collaboratively and individually to research, to analyze, and to write about, public debates regarding the conduct of science and technology;

5. To appreciate your obligations as prospective practitioners in your chosen field to the laypersons affected by your work.

ETCA364A	.NET FRAMEWORK LAB	C
		1

Course Overview:

This course teaches the programming skills that are required for developers to create Windows applications using the C# language. In the classroom, students review the basics of C# program structure, language syntax, and implementation details, and then consolidate their knowledge throughout the week as they build an application that incorporates several features of the .NET Framework. The course introduces many of the techniques and technologies employed by modern desktop and enterprise applications.

Objectives and Expected Outcome:

Students who have successfully completed this course will have full understanding of the following concepts

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#

LIST OF EXPERIMENTS

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

2. Write a program for Ad rotator Control
3. 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. Write a program for Tree view control and use various operation of Tree view control
5. Write a program to design graphical user interface and display records stored in database
6. Write a program to insert and delete the records in database
7. Write a program of Data binding using drop down list control
8. Design a interactive website for admissions in university.

ETCS555A	PYTHON PROGRAMMING LAB	C
		1

Overview: Programming and problem-solving using 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.

Objectives and Expected Outcomes:

Upon a successful completion of this course students should be able to:

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

LIST OF EXPERIMENTS

1. Develop programs to understand the control structures of python
2. Develop programs to implement list
3. Develop programs to implement Dictionary
4. Develop programs to implement tuples
5. Develop programs to implement function with stress on scoping
6. Develop programs to implement classes and objects
7. Develop programs to implement exception handling.
8. Develop programs to implement linear search and binary search.
9. Develop programs to implement insertion sort
10. Develop programs to implement bubble sort.
11. Develop programs to implement quick sort.
12. Develop programs to implement heap sort.

ETCS252A	SOFTWARE ENGINEERING LAB	C
		1

Hands-on experiments related to the course contents **ETCS202A** by performing All laboratory assignments

LIST OF EXPERIMENTS

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

ETCA368A	MAJOR PROJECT	C
		3

The major-project may be a complete hardware or a combination of hardware and software under the guidance of a Supervisor from the Department alone or jointly with a Supervisor drawn from R&D laboratory/Industry. This is expected to provide a good training for the student(s) in R&D work and technical leadership. 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/Problem Solving/Experiment as needed.
4. Final development of product/process, testing, results, conclusions and future directions.
5. Preparing a Dissertation in the standard format for being evaluated by the Department.
6. Final Seminar Presentation before a Departmental Committee.

Annexure A1

Scheme of studies as per Learning outcome Based Curriculam Framework (LOCF) and Choice based credits System (CBCS)

SOET	B.SC.(H) COMPUTER SCIENCE - YEAR 2019 (SCHEME OF STUDIES)															
YEAR	ODD SEMESTER								EVEN SEMESTER							
	SNO		COURSE CODE	COURSE TITLE	L	T	P	C	SNO		COURSE CODE	COURSE TITLE	L	T	P	C
FIRST	1	SE	ETME101A	BASICS OF MECHANICAL ENGINEERING	3	-	-	4	1	SE	BSPH132A	ELECTRICITY AND MAGNETISM	3	1	-	4
	2	OE		OPEN ELECTIVE- I	4	-	-	6	2	OE		OPEN ELECTIVE-II	4	-	-	6
	3	SE	ETEL 101A	COMMUNICATION SKILLS	4	-	-	4	3	SE	ETEL 402A	SOFT SKILLS AND PERSONALITY DEVELOPMENT	3	-	-	3
	4	CC	ETCS103A	PROGRAMMING FOR PROBLEM SOLVING	3	1	-	4	4	SE	ETCH 125A	ENVIRONMENTAL STUDIES	3	-	-	3
	5	CC	ETCS521A	COMPUTER CENTRE PLANNING & ESTABLISHMENT	3	1	-	4	5	CC	ETCS 316A	WEB TECHNOLOGIES	3	1	-	4
	6	SE	ETME151A	BASICS OF MECHANICAL ENGINEERING LAB	-	-	2	1	6	CC	ETCS112A	OBJECT ORIENTED PROGRAMMING	3	1	-	4
	7	SE	ETEL171A	COMMUNICATION SKILLS LAB	-	-	2	1	7	SE	BSPH132A	ELECTRICITY MAGNETISM LAB	-	-	2	2
	8	CC	ETCS 153A	PROGRAMMING FOR PROBLEM SOLVING LAB	-	-	2	1	8	CC	ETCA164A	WEB TECHNOLOGIES LAB	-	-	2	1
	9	CC	ETCS565A	COMPUTER CENTRE PLANNING & ESTABLISHMENT LAB	-	-	2	1	9	CC	ETCS166A	OBJECT ORIENTED PROGRAMMING LAB	-	-	2	1
				TOTAL	**17**	**2**	**8**	**26**				**TOTAL**	**19**	**3**	**6**	**28**

SECOND	1	SE	ETEC 210A	DIGITAL ELECTRONICS	3	1	-	4		1	SE	BSPH236A	ANALOG SYSTEMS AND APPLICATIONS	4	1	-	4
	2	CC	BSMA217A	REAL ANALYSIS	4	-	-	4		2	CC	BSMA137A	ALGEBRA	4	-	-	4
	3	SE	ETMA208A	NUMERICAL ANALYSIS	4	-	-	4		3	CC	ETCS 220A	ANALYSIS AND DESIGN OF ALGORITHMS	3	1	-	4
	4	CC	ETCS219A	FOUNDATION OF COMPUTER SYSTEMS	3	1	-	4		4	SE	ETMC 226A	FUNDAMENTALS OF MANAGEMENT	3	-	-	3
	5	CC	ETCS217A	DATA STRUCTURES	3	1	-	4		5	CC	ETCS307A	DATABASE MANAGEMENT SYSTEMS	3	1	-	4
	6	CC	ETCS 257A	DATA STRUCTURES LAB	-	-	2	1		6	SE	BSPH236A	ANALOG SYSTEMS AND APPLICATIONS LAB	-	-	2	2
	7	SE	ETEC 256A	DIGITAL ELECTRONICS LAB	-	-	2	1		7	CC	ETCS 355A	DATABASE MANAGEMENT SYSTEMS LAB	-	-	2	1
	8	SE	BSMA351A	NUMERICAL ANALYSIS LAB	0	0	2	1		8	CC	ETCA365A	LINUX ENVIRONMENT LAB	-	-	2	1
				TOTAL	**17**	**3**	**6**	**23**					**TOTAL**	**17**	**3**	**6**	**23**
THIRD	1	CC	ETCS304A	COMPUTER NETWORKS	3	1		4		1	CC	ETCS506A	PYTHON PROGRAMMING	3	1	-	4
	2	CC	ETCS211A	OPERATING SYSTEMS	3	1	-	4		2	CC	ETCS202A	SOFTWARE ENGINEERING	3	1	-	4
	3	CC	ETCS 206A	COMPUTER GRAPHICS	3	1		4		3	CC	ETCA324A	.Net FRAMEWORK	3	1	-	4
	4	SE	BSMA326A	OPERATIONAL RESEARCH	4	-	-	4		4	SE	ETCS505A	TECHNICAL WRITING	3	-	-	3
	5	CC	ETCS365A	COMPUTER NETWORKS LAB	-	-	2	1		5	CC	ETCA364A	.Net FRAMEWORK LAB	-	-	2	1
	6	CC	ETCS 255A	OPERATING SYSTEMS LAB	-	-	2	1		6	CC	ETCS555A	PYTHON PROGRAMMING LAB	-	-	2	1
	7	CC	ETCS258A	COMPUTER GRAPHICS LAB	-	-	2	1		7	CC	ETCS252A	SOFTWARE ENGINEERING LAB	-	-	2	1
	8		ETCS507A	SEMINAR	-	-	2	1		8		ETCA368A	MAJOR PROJECT	-	-	6	3

				TOTAL	**13**	**3**	**8**	**20**					**TOTAL**	**12**	**3**	**12**	**21**
													TOTAL CREDITS	**141**			