R25

Curriculum & Syllabus
Under Autonomy

B.Tech Computer Science and Engineering

(Internet of Things)

(Effective from 2025-26 admission batch)

Department: CSE (Internet of Things)

Curriculum Structure & Syllabus

(Effectivefrom2025-26 admission batch)

Department: CSE (IoT) ***Curriculum Structure & Syllabus (Effective from 2023-24 admission batch)***

		1st Year 1st Semester							
Sl. No.	**Broad Category**	**Category**	**Course Code**	**Course Title**	**Hours per week**				**Credits**
					L	**T**	**P**	**Total**	
A.THEORY									
1	ENGG	Multidisciplinary	YCS1001	Basic Electrical Engineering	3	0	0	3	3
2	SCI	Minor	YMT1001	Engineering Mathematics–I	3	0	0	3	3
3	SCI	Minor	YPH1001	Engineering Physics	3	0	0	3	3
4	ENGG	Major	YCS 1002	Programming for Problem Solving	3	1	0	4	4
5	HUM	Ability Enhancement Course	YED1003	Indian Knowledge System	2	0	0	2	1
B.PRACTICAL									
6	ENGG	Multidisciplinary	YCS 1101	Basic Electrical Engineering Laboratory	0	0	3	3	1.5
7	ENGG	Skill Enhancement	YCS1103	Workshop & Manufacturing Practices Laboratory	0	0	3	3	1.5
8	ENGG	Major	YCS1102	Programming for Problem Solving Laboratory	0	0	3	3	1.5
9	SCI	Minor	YPH1101	Engineering Physics Laboratory	0	0	3	3	1.5
C.MANDATORYACTIVITIES/COURSES(Non-CGPA)									
10	MC	Mandatory Course	YCS1502	Seminar/Group Discussion	0	0	0	0	1
11	MC	Mandatory Course	YCS1503	SkillX	0	0	0	0	1
Total of Theory, Practical and Mandatory Activities /Courses								27	20

HUM:Humanities;ENGG: Engineering;SCI:Science;PRJ:Project;MC:**MandatoryActivities/Courses**

1st Year2nd Semester									
Sl. No.	Broad Category	Category	Course Code	Course Title	Hours per week				Credits
					L	T	P	Total	
A.THEORY									
1	ENGG	Minor	YCS2001	Basic Electronics Engineering	3	0	0	3	3
2	ENGG	Major	YCS2002	Data Structures	3	1	0	4	4
3	SCI	Multidisciplinary	YCH2001	Engineering Chemistry	3	0	0	3	3
4	ENGG	Minor	YMT2001	Engineering Mathematics–II	3	1	0	4	4
5	HUM	Common value added course	YED2001	Professional Communication	1	1	0	2	1
B.PRACTICAL									
6	ENGG	Major	YCS2102	Data Structures Laboratory	0	0	3	3	1.5
7	ENGG	Skill Enhancement	YCS2101	Basic Electronics Laboratory	0	0	3	3	1.5
8	HUM	Minor	YED2101	Professional Communication Laboratory	0	0	2	2	1
9	ENGG	Skill Enhancement	YCS2301	Engineering Graphics & Design Laboratory	0	0	3	3	1.5
10	SCI	Multidisciplinary	YCH2101	Engineering Chemistry Laboratory	0	0	3	3	1.5
11	HUM	Ability Enhancement	YCS2505/ JSC2503	Life Skills / Communicative English Proficiency	0	0	1	1	1
C.MANDATORYACTIVITIES/COURSES(Non-CGPA)									
11	HUM	Ability Enhancement Course	YCS2502	Seminar / Group Discussion	0	0	1	1	1
12	MC	Mandatory Course	YCS2501	NSS/Physical Activities /Meditation &Yoga/ Photography	0	0	1	1	1
13	MC	Mandatory Course	YCS2504	SkillX	0	0	1	1	1
Total of Theory, Practical and Mandatory Activities /Courses								33	23

2nd Year3rd Semester									
Sl. No.	Broad Category	Category	Course Code	Course Title	Hours per week				Credits
					L	T	P	Total	
A.THEORY									
1	ENGG	Minor	YCS3005	Probability and Statistics	3	0	0	3	3
2	ENGG	Minor	YCS3001	Digital Logic and Electronics	3	0	0	3	3
3	ENGG	Major	YCS3002	Computer Organization and Architecture	3	0	0	3	3
4	ENGG	Major	YCS3003	Design and Analysis of Algorithms	3	0	0	3	3
5	SCI	Minor	YCS3004	Computational Biology	3	0	0	3	3
B.PRACTICAL									
6	ENGG	Minor	YCS3101	Digital Logic and Electronics Laboratory	0	0	3	3	1.5
7	ENGG	Major	YCS3102	Computer Organization and Architecture Laboratory	0	0	3	3	1.5
8	ENGG	Major	YCS3103	Design and Analysis of Algorithms Laboratory	0	0	3	3	1.5
9	ENGG	Skill Enhancement Course	YCS3104	IT Workshop Laboratory(SciLab/MATLAB/C++)	0	1	3	4	2.5
C.MANDATORYACTIVITIES/COURSES(Non-CGPA)									
10	HUM	Ability Enhancement Course	YCS3501	Technical Seminar Presentation	0	0	1	1	1
11	MC	Mandatory Course	YCS3503	Ethical Hacking Concept	0	0	1	1	1
12	MC	Mandatory Course	YCS3502	SkillX	0	0	0	0	1
Total of Theory, Practical and Mandatory Activities / Courses								28	22

UNIVERSITY

2nd Year 4th Semester									
Sl. No.	Broad Category	Category	Course Code	Course Title	Hours per week				Credits
					L	T	P	Total	
A.THEORY									
1	ENGG	Minor	YCS4005	Discrete Mathematics	3	0	0	3	3
2	ENGG	Major	YCS4001	Operating Systems	3	0	0	3	3
3	ENGG	Major	YCS4002	Formal Language and Automata Theory	3	0	0	3	3
4	ENGG	Major	YCS4003	Object Oriented Programming using Java	3	0	0	3	3
5	HUM	Ability Enhancement Course	YBB4001	Principles of Management	2	0	0	2	2
B.PRACTICAL									
7	ENGG	Major	YCS4101	Operating Systems Laboratory	0	0	3	3	1.5
8	ENGG	Minor	YCS4102	Numerical Methods Laboratory	0	0	3	3	1.5
9	ENGG	Major	YCS4103	Object Oriented Programming using Java Laboratory	0	0	3	3	1.5
10	ENGG	Major	YCS4104	Programming Using Python Laboratory	0	0	3	3	1.5
C.MANDATORYACTIVITIES/COURSES(Non-CGPA)									
11	HUM	Ability Enhancement Course	YCS4501	Soft Skill & Aptitude	1	0	0	1	1
12	MC	Mandatory Course	YED4501	Foreign Language	1	0	0	1	1
13	MC	Mandatory Course	YCS4502	SkillX	0	0	1	1	1
Total of Theory, Practical and Mandatory Activities /Courses								26	20

UNIVERSITY

3rd Year 5th Semester									
Sl. No.	**Broad Category**	**Category**	**Course Code**	**Course Title**	**Hours per week**				**Credits**
					L	**T**	**P**	**Total**	
A.THEORY									
1	ENGG	Major	YCS 5006	Real Time System	3	0	0	3	3
2	ENGG	Major	YCS 5002	Database Management Systems	3	0	0	3	3
3	ENGG	Minor	YCS 5002	Computer Networks	3	0	0	3	3
4	ENGG	Major	YCS 5004A YCS 5004B YCS 5004C	Compiler Design Cryptography and Network Security Computer Graphics	3	0	0	3	3
5	HUM	Minor	YED 5001	Economics for Engineers	2	0	0	2	2
B.PRACTICAL									
6	ENGG	Major	YCS 5106	Real Time System Laboratory	0	0	3	3	1.5
7	ENGG	Major	YCS 5102	Database Management Systems Laboratory	0	0	3	3	1.5
8	ENGG	Major	YCS 5102	Computer Networks Laboratory	0	0	3	3	1.5
9	PRJ	Skill Enhancement	YCS 5201	Internship	0	0	2	2	1.5
C.MANDATORYACTIVITIES/COURSES(Non-CGPA)									
10	MC	Mandatory Course	YCS 5501	Behavioral and Interpersonal skills	3	0	0	3	1
11	MC	Mandatory Course	YCS5502	SkillX	0	0	0	0	1
Total of Theory, Practical and Mandatory Activities /Courses								28	20

3rd Year 6th Semester									
Sl. No.	**Broad Category**	**Category**	**Course Code**	**Course Title**	**Hours per week**				**Credits**
					L	**T**	**P**	**Total**	
A.THEORY									
1	ENGG	Major	YCS 6009	IoT Communication Protocols	3	1	0	4	4
2	ENGG	Major	YCS 6010	Embedded Systems	3	0	0	3	3
3	ENGG	Major	YCS 6011	Programming Languages for IoT	3	0	0	3	3
4	ENGG	Major	YCS 6004C YCS 6004F YCS 6004G	Cloud Computing Software Engineering IT Security Compliance and Forensics	3	0	0	3	3
5	ENGG	Minor	YCS 6012	Industrial IoT	3	0	0	0	3
B.PRACTICAL									
6	ENGG	Major	YCS 6109	IoT Communication Protocols Laboratory	0	0	3	3	1.5
7	ENGG	Major	YCS 6110	Embedded Systems Laboratory	0	0	3	3	1.5
8	ENGG	Major	YCS 6111	Programming Languages for IoT Laboratory	0	0	3	3	1.5
C.MANDATORYACTIVITIES/COURSES(Non-CGPA)									
9	MC	Mandatory Course	YCS6501	Innovation, Entrepreneurship & IPR	3	0	0	3	1
10	MC	Mandatory Course	YCS6502	SkillX	0	0	0	0	1
TotalofTheory,PracticalandMandatoryActivities /Courses								24	20.5

4th Year 7th Semester									
Sl. No.	**Broad Category**	**Category**	**Course Code**	**Course Title**	**Hours per week**				**Credits**
					L	**T**	**P**	**Total**	
A.THEORY									
1	ENGG	Major	YCS7006	IoT Cloud Processing and Analytics	3	1	0	4	4
2	ENGG	Major	YCS7002F YCS7002G YCS7002H	Data Science for Internet of Things Artificial Intelligence Fog Computing	3	0	0	3	3
3	ENGG	Major	YCS7003G YCS7003H YCS7003I	Blockchain Technology Ad-Hoc and Sensor Networks Augmented Reality & Virtual Reality	3	0	0	3	3
4			YCS 7004D YCS 7004E YCS 7004F	Advanced Algorithm Neural Networks and Deep Learning OptimizationTechniques	3	0	0	3	3
5	HUM	Minor	YBB7001	Human Resource Development and Organizational Behavior	2	0	0	2	1
B.PRACTICAL									
6	ENGG	Major	YCS 7106	IoT Cloud Processing and Analytics Laboratory	0	0	3	3	1.5
7	ENGG	Major	YCS 7102 F YCS 7102 G YCS 71 02 H	Data Science Laboratory Artificial Intelligence Laboratory Cloud Computing Laboratory	0	0	3	3	1.5
8	ENGG	Major	YCS 7201	Project I	0	0	3	3	2.5
C.MANDATORY ACTIVITIES/COURSES(Non-CGPA)									
9	MC	Mandatory Course	YCS7501	Constitution of India	3	0	0	3	1
10	MC	Mandatory Course	YCS7502	SkillX	0	0	0	0	1
Total of Theory, Practical and Mandatory Activities/Courses								29	19.5

4thYear8thSemester									
Sl.No.	**Broad Category**	**Category**	**Course Code**	**Course Title**	**Hours per week**				**Credits**
					L	**T**	**P**	**Total**	
A.THEORY									
1	ENGG	Major	YCS 8001G YCS 8001H YCS 8001I	Sensors And Devices Data Analytics Soft Computing	3	1	0	3	4
2	ENGG	Major	YCS 8002G YCS 8002H YCS 8002I	Quantum Computing Bio-informatics Robotics	3	1	0	3	4
B.PRACTICAL									
3	PRJ	Skill Enhancement	YCS 8201	Project	0	0	18	18	9
4	PRJ	Internship	PR891	Grand Viva	0	0	6	6	3
C.MANDATORYACTIVITIES/COURSES(Non-CGPA)									
5	MC	Mandatory Course	YED 8501	Essence of Indian Knowledge Tradition	0	0	3	3	1
6	MC	Mandatory Course	YCS8502	Skillx	0	0	0	0	1
Total of Theory, Practical and Mandatory Activities/Courses								33	20

Total Credit = 165

Semester1Curriculum and Syllabus

Course Code	YCS1001			
Course Title	Basic Electrical Engineering			
Category	Engineering Science			
LTP &Credits	L	T	P	Credits
	3	0	0	3
Total Contact Hours	36			
Pre-requisites	None			

Learning Objective:

In this course, students will learn about the fundamentals of electrical circuits, focusing on both DC and AC circuits, transformers, and rotating machines.

Course Outcome:

CO1: Understand and remember the working principles of basic electrical circuits, power distribution, and safety measures.
CO2: Analyze the functioning of DC and AC circuits.
CO3: Understand and remember the basic principles of transformers and rotating machines.
CO4: Analyze and evaluate the performance of electrical circuits, transformers, and rotating machines through laboratory experiments and simulations.
CO5: Design basic electrical circuits and systems, considering safety, efficiency, and economic factors.

Course Content:

Module 1: DC Circuits Fundamentals [9L]

- Electric circuits: Linear circuit, Non-linear circuit, Bilateral circuit, Unilateral circuit, Dependent source, Node, Branch, Active and Passive elements, Kirchhoff's laws, Source equivalence and conversion.
- Network Theorems: Superposition theorem, Thevenin's theorem, Norton's theorem, Maximum power transfer theorem.

Module 2: AC Circuits Fundamentals [9L]

- Sinusoidal quantities: Average and RMS values, Peak factor, Form factor, Phase and Phase difference, Concept of phasor diagram, V-I relationship in R, L, C circuits, R-L-C circuits in series and parallel with phasor diagrams, Impedance and Admittance, Impedance triangle and Power triangle, Power factor.
- Three-phase balanced circuits, Concept of three-phase power measurement.

Module 3: Single-Phase Transformer [5L]

- Single-phase transformer: Idea on constructional parts, Classifications, Working principle. Numerical on EMF equation, Phasor diagram.

Module 4: Electrical Rotating Machines [8L]

- DC Machines: Constructional features, Classifications, Working principle of Generator and Motor. Simple Numerical on voltage equation.
- Three-phase Induction Motor: Basic concept of Three Phase circuit and Production of Rotating Magnetic Field, Working principle of Three Phase Induction Motor and Torque-Speed Characteristics (concept only).

Module 5: Electrical Installations [5L]

- Power generation to distribution through overhead lines and underground cables with single-line diagram. Earthing of electrical equipment, Basic accessories: MCB, MCCB, ELCB, SFU, Megger.

Text/Reference Books:

1. D.P. Kothari and I.J. Nagrath, "Basic Electrical Engineering", Tata-McGrawHill.
2. V. Mittle and A. Mittal, "Basic Electrical Engineering", Tata-McGrawHill.
3. E. Hughes, "Electrical and Electronics Technology", PHI/Pearson Education.
4. C.L. Wadhwa, "Basic Electrical Engineering", Pearson Education.

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YMT1001			
Course Title	Engineering Mathematics I			
Category	SCI Minor			
LTP & Credits	L	T	P	Credits
	3	0	0	3
Total Contact Hours	36			
Pre-requisites	None			

Learning Objectives:

The goal of this course is to achieve conceptual understanding and to retain the best traditions of traditional calculus. The syllabus is designed to provide the basic tools of calculus mainly for the purpose of modelling the engineering problems mathematically and obtaining solutions. This is a foundation course which mainly deals with topics such as single variable and multivariable calculus and plays an important role in the understanding of science, engineering, economics and computer science, among other disciplines.

Course Outcome:

CO1: To apply differential and integral calculus to notions of curvature and to improper integrals. Apart from some other applications they will have a basic understanding of Beta and Gamma functions.

CO2: The fallouts of Rolle's Theorem that is fundamental to application of analysis to Engineering problems.

CO3: The tool of power series and Fourier series for learning advanced Engineering Mathematics.

CO4: To deal with functions of several variables that is essential in most branches of engineering.

CO5: To acquaint the student with mathematical tools needed in evaluating multiple integrals and their usage

Course Content:

Module 1: Basic Calculus: [6L]

Curvature, evolutes and involutes; Evaluation of definite and improper integrals; Beta and Gamma functions and their properties; Applications of definite integrals to evaluate surface areas and volumes of revolutions. .

Module 2: Single-variable Calculus (Differentiation): [6L]

Rolle's Theorem, Mean value theorems and applications; Extreme values of functions; Linear approximation; Indeterminate forms and L' Hospital's rule..

Module 3: Sequences and series: [8L]

Limits of sequence of numbers, Calculation of limits, Infinite series; Tests for convergence; Power series, Taylor and Maclaurin series; Taylor theorem, convergence of Taylor series, error estimates..

Module 4: Multivariable Calculus (Differentiation): [6L]

Limit, continuity and partial derivatives, directional derivatives, gradient, total derivative; Tangent plane and normal line; Maxima, minima and saddle points; Method of Lagrange multipliers.

Module 5: Multivariable Calculus (Integration): [10L]

Multiple Integration: Double integrals (Cartesian), change of order of integration in double integrals, Change of variables (Cartesian to polar), Applications: areas and volumes, Center of mass and Gravity (constant and variable densities); Triple integrals (Cartesian), orthogonal curvilinear coordinates, Simple applications involving cubes, sphere and rectangular parallelepipeds; Scalar line integrals, vector line integrals, scalar surface integrals, vector surface integrals, Gradient, curl and divergence, Theorems of Green, Gauss and Stokes..

Text / Reference Books:

1. E. Kreyszig, "Advanced Engineering Mathematics (9th Ed)", John Wiley & Sons, 2006.

2. B. V. Ramana, “Higher Engineering Mathematics”, Tata McGraw Hill New Delhi, 11th Reprint, 2010.
3. T. Veerarajan, “Engineering Mathematics for first year”, Tata McGraw-Hill, New Delhi, 2008.
4. B. S. Grewal, “Higher Engineering Mathematics (36th Ed)”, Khanna Publishers, 2010.
5. N. P. Bali and M. Goyal, “A textbook of Engineering Mathematics”, Laxmi Publications, Reprint, 2008.

PO-CO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YPH1001			
Course Title	Engineering Physics			
Category	Basic Science			
LTP & Credits	L	T	P	Credits
	3	0	0	3
Total Contact Hours	36			
Pre-requisites	None			

Learning Objective:

The objective of the course is to provide students with a sufficient understanding of physics' foundational ideas and their potential applications. Understanding the fundamentals of physics will enable engineers to better comprehend the instruments and methods employed in the field and lay the groundwork for introducing novel ideas. It can also raise awareness of the crucial part that engineering and science play in the creation of new technologies. Additionally, it provides the required exposure to the practical aspects, which is a vital component of studying the sciences.

Course Outcome:

CO1: To understand and remember the basic principles of Classical Mechanics, Relativistic Mechanics, Quantum Mechanics, Statistical Mechanics, and microscopic phenomena.

CO2: To understand the working principles of laser, optical fiber, and holography amplifier and analyze their various applications to enhance knowledge in modern optics.

CO3: To learn the basic principles of electromagnetic induction, dielectric properties, and magnetic properties of materials.

CO4: To understand and apply knowledge in new technologies.

CO5: To apply the principles of physics to solve engineering problems and analyze real-world phenomena.

Course Content:

Module 1: Mechanics [8L]

- Representation of vector, scalar, and vector fields, partial derivative of vector, gradient of scalar field, divergence, and curl of vector field.
- Friction, conservation laws, rigid body, moment of inertia, acceleration of rigid body, Mass-energy Equivalence Concept of photon.

Module 2: Quantum Theory [8L]

- Blackbody Radiation spectrum, Wein's law, Rayleigh–Jeans law, Quantum theory of radiation, Wave mechanics, wave-particle duality, De Broglie waves, Bohr's quantization rules, Phase and group velocities, Davission-Germer Experiment, Heisenberg Uncertainty Principle, Wave function and its significance, Schrodinger's wave equation.

Module 3: Laser, Fiber Optics [6L]

- Spontaneous and stimulated emission of radiation, Population inversion, Einstein's coefficients, Basic components of laser system, Construction and working of Ruby laser, He-Ne lasers, Laser Applications. Basic ideas of Fiber optics.

Module 4: Statistical Mechanics and Applications [6L]

- Introduction to Statistical mechanics, Concept of energy levels and energy states. Classical limits of quantum statistics, Concept of Fermi level. Fermi level in metals, Fermi level for intrinsic and extrinsic semiconductors (pictorial representations on temperature dependence and doping concentration viz. p-type, n-type).

Module 5: Electromagnetic Induction [3L]

- Magnetic flux, Faraday's law of electromagnetic induction, electromotive force, Ampere's circuital law, Maxwell's equation.

Module 6: Dielectrics [3L]

- Types of dielectric, relation between dielectric constant and electric susceptibility, polarizability, Clausius-Mossotti Equation, application of dielectric materials.

Module 7: Magnetic Properties of Materials [2L]

- Magnetic flux density, magnetic permeability, magnetic susceptibility, classification of magnetic materials, diamagnetic materials, paramagnetic materials, Curie law.

Text/Reference Books:

1. Beiser, "Concepts of Modern Physics", McGraw Hill India.
2. D.K. Bhattacharya and P. Tandon, "Engineering Physics", Oxford India.
3. Lal and N. Subramanyam, "A Textbook Of Optics", S. Chand & Co.
4. Dominic I. and A. Nahari, "A TextBook of Engineering Physics", Owl Book Publishers.
5. E. Hecht, "Optics", Pearson Education.
6. N. Mehta, "Applied Physics for Engineers", PHI Ltd.
7. J.C Palais, "Fiber Optic Communications", Pearson Education.
8. K. Pandey and S. Chathurvedi, "Engineering Physics", Cengage Learning.

9. J. Philip, “A textbook of Engineering Physics”, Educational Publishers.

10. B. Premlet, “Engineering Physics”, McGraw Hill India..

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS 1002			
Course Title	Programming for Problem Solving			
Category	ENGG Major			
LTP & Credits	L	T	P	Credit
	3	1	0	4
Total Contact Hours	48			
Pre-requisites	None			

Learning Objective:

Algorithmic skill is a fundamental skill in modern times, and this course provides the students with the foundations of computational problem solving. The course emphasizes on principles and methods rather than on systems and tools.

Course Outcome:

CO1: Understand the basic model of computation
CO2: Apply algorithmic thinking to understand, and solve problems
CO3: Design and implement algorithms for a given problem
CO4: Implement algorithms using a high-level programming language
CO5: Analyze and evaluate algorithms for efficiency and correctness

Course Content:

Module 1: Problem Solving and Algorithmic Thinking [6L]

Overview - problem definition, logical reasoning. Flowcharts - symbols used, examples. Algorithm - definition, practical examples, properties, representation, algorithms vs programs. Elementary concepts about time complexity

Module 2: Introduction to Computation [10L]

Introduction to Computing – Computer Systems-Hardware and Software, Computer Languages, Algorithm, Flowchart, Representation of Algorithm and Flowchart with examples. Introduction to

C– History of C, Features of C, Structure of C Program, Character Set, C Tokens-Keywords, Identifiers, Constants, Variables, Data types, Operators.

.Module 3: Function of C [8L]

Functions-Designing Structured Programs, Types of Functions-User defined functions, Standard functions, Categories of functions, Parameter Passing techniques, Recursion.

Module 4: Array and String [8L]

Arrays- Declaration and Initialization, One dimensional Array, Two dimensional Arrays. Strings-Declaration and Initialization, String Input / Output functions, String manipulation functions.

Module 5: Pointers and Structure [9L]

Pointers- Introduction, Definition and Declaration of pointers, address operator, Pointer variables, Pointers with Arrays. Structures- Introduction, Declaration and Initialization, Array of Structures, Unions.

Module 5: File Handling in C [6L]

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

Text/Reference Books:

1. D.D. Riley and K. A. Hunt, \Computational Thinking for the Modern Problem Solver", CRC Press.
2. P. F. Luccio, \Computational Thinking: First Algorithms, then Code" Springer.
3. S. S. Sastry, \Introductory Methods of Numerical Analysis" Prentice-Hall of India.
4. R. G. Dromey, \How to Solve it by Computer" Prentice-Hall.

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YED1005			
Course Title	Indian Knowledge System			
Category	Core			
LTP & Credits	L	T	P	Credits
	1	1	0	1
Total Contact Hours	24 hrs			
Pre-requisites	None			

Learning Objective(s):

To facilitate students with the concepts of Indian traditional knowledge and to make them understand the importance of roots of the knowledge system. To make students understand traditional knowledge and analyze it and apply it to their day-to-day life.

Course Outcomes:

CO1: Provide an overview of the concept of the Indian Knowledge System and its importance.
CO2: Appreciate the need and importance of protecting traditional knowledge.
CO3: Recognize the relevance of Traditional knowledge in different domains.
CO4: Establish the significance of Indian Knowledge systems in the contemporary world.
CO5: Evaluate the impact of Indian Knowledge Systems on global perspectives.

Course Content:

Module 1: Introduction to Indian Knowledge Systems (IKS): [8L]

Overview, Vedic Corpus, Philosophy, Character scope and importance, traditional knowledge vis-a-vis indigenous knowledge, traditional knowledge vs. western knowledge.

Module 2: Traditional Knowledge in Humanities and Sciences: [8L]

Linguistics, Number and measurements- Mathematics, Chemistry, Physics, Art, Astronomy, Astrology, Crafts and Trade in India and Engineering and Technology.

Module 3: Traditional Knowledge in Professional domain: [8L]

Town planning and architecture- Construction, Health, wellness and Psychology-Medicine, Agriculture, Governance and public administration, United Nations Sustainable development goals.

Text / Reference Books:

1. Introduction to Indian Knowledge System - concepts and applications, B Mahadevan, Vinayak Rajat Bhat, Nagendra Pavana R N, 2022, PHI Learning Private Ltd, ISBN-978-9391818-21-0
2. Traditional Knowledge System in India, Amit Jha, 2009, Atlantic Publishers and Distributors (P) Ltd., ISBN-13: 978-8126912230
3. Knowledge Traditions and Practices of India, Kapil Kapoor, Avadesh Kumar Singh, Vol. 1, 2005, DK Print World (P) Ltd., ISBN 81-246-0334

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS 1103			
Course Title	Workshop/Manufacturing Practices Laboratory			
Category	ENGG Skill Enhancement			
LTP & Credits	L	T	P	Credits
	0	0	3	1.5
Total Contact Hours	36			
Pre-requisites	Physics & Mathematics (10+2 Level)			

Course Outcomes:

After successful completion of this course, learners will be able to:

CO1: Identify and operate various hand tools related to a variety of manufacturing operations.
CO2: Safely fabricate simple components with their own hands.
CO3: Gain practical knowledge of the dimensional accuracies and tolerances applicable for different manufacturing processes.
CO4: Produce small devices of their interest for project or research purposes.
CO5: Apply safety standards and regulations in workshop environments to ensure the well-being of oneself and others.

Course Content:

(i) Theoretical discussion & videos: 3P

1. Manufacturing Methods - casting, forming, machining, joining, advanced manufacturing methods
2. Fitting operations & power tools
3. Carpentry

4. Welding (arc welding & gas welding), brazing
5. Electrical & Electronics
6. Metal casting
7. CNC machining, Additive manufacturing
8. Plastic molding & Glass Cutting

(ii) Workshop Practice:

Module 1: Machine shop 6P Typical jobs that may be made in this practice module: i. To make a pin from a mild steel rod in a lathe. ii. To make rectangular and vee slot in a block of cast iron or mild steel in a shaping and / or milling machine.

Module 2: Fitting shop 6P Typical jobs that may be made in this practice module: i. To make a Gauge from MS plate.

Module 3: Carpentry 6P Typical jobs that may be made in this practice module: i. To make wooden joints and/or a pattern or like.

Module 4 : Welding shop (Arc welding 3P + gas welding 3P) 3P Typical jobs that may be made in this practice module: i. ARC WELDING (3P): To join two thick (approx 5mm) MS plates by manual metal arc welding. ii. GAS WELDING (3P): To join two thin mild steel plates or sheets by gas welding.

Module 5 : Electrical & Electronics 3P House wiring, soft soldering

Module 6: Smithy 3P Typical jobs that may be made in this practice module: R21 B. Tech (ME) i. A simple job of making a square rod from a round bar or similar. For further study (Optional)

Module 7: Casting 3P Typical jobs that may be made in this practice module: i. One/ two green sand molds to prepare, and a casting be demonstrated.

Module 8: Plastic molding & Glass Cutting 3P Typical jobs that may be made in this practice module: i. For plastic molding, making at least one simple plastic component should be made. ii. At least one sample shape on glass should be made using laser cutting machine. Examinations could involve the actual fabrication of simple components, utilizing one or more of the techniques covered above.

Text Books:

1. Hajra Choudhury S.K., Hajra Choudhury A.K., and Nirjhar Roy S.K., "Elements of Workshop Technology", Vol. I 2008 and Vol. II 2010, Media promoters and publishers private limited, Mumbai.
2. Rao P.N., "Manufacturing Technology", Vol. I and Vol. II, Tata McGrawHill House, 2017.

Reference Books:

1. Gowri P., Hariharan and A. Suresh Babu, Manufacturing Technology – I, Pearson Education, 2008.
2. Roy A. Lindberg, "Processes and Materials of Manufacture", 4th edition, Prentice Hall India, 1998.
3. Kalpakjian S. and Steven S. Schmid, Manufacturing Engineering and Technology, 4th edition, Pearson Education India Edition, 2002.

4. Manufacturing Science by A. Ghosh and A.K. Mallick, Wiley Eastern.
5. Principles of Metal Cutting/Principles of Machine Tools by G.C. Sen and A. Bhattacharya, New Central Book Agency, Kolkata.

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course code	YCS1102			
Course Title	Programming for Problem solving Laboratory			
Category	ENGG Major			
LTP & Credits	L	T	P	Credit
	0	0	3	1.5
Total Contact Hours	36			
Pre-requisites	Basic Understanding of Computer			

Learning Objective:

The course is oriented to those who want to advance structured and procedural programming understanding and to improve C programming skills. The major objective is to provide students with an understanding of code organization and functional hierarchical decomposition using complex data types.

Course Outcomes:

CO1: Learn and understand the DOS system commands and familiarize with C programming environment.

CO2: Learn and translate the algorithms into simple programs and understand the flowchart design and test.

CO3: Understand and implement conditional branching, iteration and recursion.

CO4: Apply and analyze various C programs with Arrays, Pointers, Structures, Union along with functions.

CO5: Apply programming to solve matrix addition and multiplication problems and understand the file handling.

Course Content:

Suggestive List of Experiments:

1. Familiarization with basic DOS commands and programming design with the help of Flowcharts Using Raptor. [1 day]

2. Familiarization with C programming environment, Variable types and type Conversions, Simple Computational problems using arithmetic expressions. [1 day]
3. Branching and logical expressions, Problems involving if-then-else structures. [1 day]
4. Loops, while and for loops, Iterative problems e.g., sum of series, patterns print. [2 days]
5. 1D Arrays: searching, sorting, 1D Array manipulation, 2D arrays and Strings, Matrix problems, String operations. [2 days]
6. Functions, call by value, Simple functions implementations, function recursion. [2 days]
7. Pointers, structures and dynamic memory allocation, Union. [2 days]
8. File handling, file reading, writing, copying etc. [1 day]

Text/Reference Books:

1. B. W. Kerninghan& D. M. Ritchie, "The C Programming Language (16th Ed.)", PHI/Pearson Education.
2. Y. Kanetkar, "Let us C (15th Ed.)", BPB Publication.
3. E. Balagurusamy, "Programming in ANSI C (15th Ed.)", Tata-McGraw Hill.
4. K. R. Venugopal& S. R. Prasad, "Mastering C (7th Ed.)", Tata-McGraw Hill.
5. R. Thareja, "Introduction to C Programming (4th Ed.)", Oxford University Press.

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YPH1101			
Course Title	Engineering Physics Laboratory			
Category	Basic Science			
LTP & Credits	L	T	P	Credits
	0	0	3	1.5
Total Contact Hours	36			
Pre-requisites	None			

Learning Objective:

The objective of this course is to revise the basic concepts of physics through a standard set of experiments to correlate them with the corresponding theory.

Course Outcomes:

CO1: To discover an idea of different measurements and errors.
CO2: To understand and apply basic laws of physics and experiments.
CO3: To practice and generate experimental skills in different areas of physics and applications.
CO4: To analyze experimental data using appropriate statistical methods and tools to draw meaningful conclusions and make predictions.
CO5: To communicate experimental procedures, observations, and results effectively through written reports and oral presentations.

Course Content:

Suggestive List of Experiments:

1. General idea about Measurements and Errors: i) Error estimation using Slide calipers/Screw-gauge/traveling microscope for one experiment. ii) Proportional error calculation using Carey Foster Bridge. [2 days]
2. Determination of wavelength of light by Newton's ring method. [1 day]
3. Determination of wavelength of light by Laser diffraction method. [1 day]
4. Determination of Planck's constant using photoelectric cell. [1 day]
5. Verification of Bohr's atomic orbital theory through Frank-Hertz experiment. [2 days]
6. Determination of Stefan's constant. [1 day]
7. Determination of bandgap of a semiconductor. [1 day]
8. Study of dispersive power of material of a prism. [1 day]
9. Measurement of nodal and antinodal points along transmission wire and measurement of wavelength. [1 day]
10. Determination of wavelength of light by Fresnel's bi-prism method. [1 day]

Text/Reference Books:

1. B.L. Flint and H.T. Worsnop, "Advanced Practical Physics for Students", Asia Publishing House.
2. M. Nelson and J.M. Ogborn, "Advanced Level Physics Practicals", Heinemann Educational Publishers.
3. S. Panigrahi and B. Mallick, "Engineering Practical Physics", Cengage Learning.

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS1101			
Course Title	Basic Electrical Engineering Laboratory			
Category	Engineering Science			
LTP & Credits	L	T	P	Credits
	0	0	3	1.5
Total Contact Hours	36			
Pre-requisites	None			

Learning Objective:

In this course, students will learn about the basic electrical components, machinery, instruments, and safety measures.

Course Outcomes:

CO1: Identify and apply common electrical equipment and instruments.
CO2: Develop electric networks using various components and analyze the circuit behavior.
CO3: Apply and analyze the basic characteristics of transformers and electrical machines.
CO4: Demonstrate an understanding of the characteristics and operation of different types of electrical machines.
CO5: Analyze and interpret data from electrical experiments, draw conclusions, and make recommendations based on the findings.

Course Content:

List of Experiments:

1. Basic safety precautions – Earthing, Introduction to measuring instruments – Voltmeter, Ammeter, Multimeter, Wattmeter, Real-life Resistor, Capacitor, Inductor.
2. Verification of Thevenin's and Norton's Theorem.
3. Verification of Superposition and Maximum Power Transfer Theorem.
4. Characteristics of Fluorescent, Tungsten, and Carbon filament lamps.
5. Study of R-L-C series circuit.
6. Three-phase Power measurement with two wattmeter methods.
7. Demonstration of cut-out sections of machines: DC Machine (commutator-brush arrangement), Induction Machine (squirrel cage rotor).
8. Measurement of primary and secondary voltage and current of a single-phase transformer – Open Circuit and Short Circuit Test.
9. Starting, Reversing, and speed control of DC shunt motor.
10. Torque-Speed characteristics of DC Machine.
11. Torque-Speed characteristics of Three-phase Induction Motor.
12. Test on a single-phase Energy Meter.
13. Innovative experiments.

Text/Reference Books:

1. D.P. Kothari and I.J. Nagrath, "Basic Electrical Engineering," Tata-McGraw Hill.
2. V. Mittle and A. Mittal, "Basic Electrical Engineering," Tata-McGraw Hill.
3. E. Hughes, "Electrical and Electronics Technology," Pearson.
4. C.L. Wadhwa, "Basic Electrical Engineering," Pearson Education.

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

JIS UNIVERSITY

Semester 2 Curriculum and Syllabus

Course Code	YCS2001			
Course Title	Basic Electronics			
Category	SCI Multi Displ			
LTP & Credits	L	T	P	Credits
	3	0	0	3
Total Contact Hours	36			
Pre-requisites	None			

Learning Objective:

In this course, students will learn about the fundamental behavior and principle of operations of various electronic devices and circuits. At the end of the course, students will be able to design useful electronic subsystems like rectifiers, amplifiers, etc.

Course Outcome:

CO1: To understand and remember the principle of operation of semiconductor devices.
CO2: To understand and analyze the operations of P-N junction diodes, bipolar and field-effect transistors, and solve design problems.
CO3: To understand and remember the principle of working of operational amplifiers and demonstrate its various applications.
CO4: Design and analyze electronic circuits incorporating semiconductor devices and operational amplifiers to meet specified requirements.
CO5: Troubleshoot and optimize electronic circuits using diagnostic tools and techniques.

Course Content:

Module 1: Fundamentals of Semiconductor **[6L]**

- Conductors, Insulators, and Semiconductors
- Energy band diagrams
- Fermi Dirac function, Fermi level
- Intrinsic and Extrinsic (p-type and n-type) semiconductors
- Drift and Diffusion current
- Mass action law
- Conductivity in semiconductor
- Einstein relationship in semiconductor
- Numerical problems

Module 2: P-N Junction Diode and its Applications **[8L]**

- P-N junction diode, Characteristics, and parameters
- Diode equation, V-I characteristics
- P-N junction breakdown: conditions, avalanche, and zener breakdown
- Zener diode and characteristics
- Diode half-wave and full-wave rectifiers (centre tapped and bridge) circuits and operation,

ripple factor, efficiency, reduction of ac ripples using filter circuit (Qualitative analysis)
- Design of diode clipper and clamper circuit
- Application of Zener diode in regulator circuit
- Numerical problems

Module 3: Bipolar Junction Transistor [6L]

- BJT operation: PNP and NPN transistors
- Transfer characteristics, Current conduction mechanism
- Common Emitter, Common Base, Common Collector configurations and Static characteristics
- Junction biasing condition for active, saturation, and cut-off modes
- DC load line and Quiescent point
- BJT biasing, BJT as an amplifier
- Numerical problems

Module 4: Field Effect Transistor [6L]

- Classification of field-effect transistors: JFET, MOSFET
- Operating principle of JFET
- Drain and Transfer characteristics of JFET (n-channel and p-channel)
- CS, CG, CD configurations
- Relation between JFET parameters
- FET as an amplifier
- E-MOSFET (n-channel and p-channel), D-MOSFET (n-channel and p-channel)
- Numerical Problems

Module 5: Feedback and Operational Amplifier [8L]

- Concept of feedback: Positive and Negative feedback, Gain with feedback, Feedback topologies
- Operational Amplifier: Ideal characteristics, Non-ideal characteristics of op-amp – offset voltages, bias current, offset current, slew rate, CMRR, and bandwidth
- Inverting and Non-inverting amplifier
- Concept of virtual ground and virtual short
- Applications of op-amp: Adder, Differential Amplifier, Differentiator, Integrator
- Numerical problems

Module 6: Electronic Instruments and Measurements [2L]

- Basics of measurement
- Cathode-Ray Oscilloscope and Digital-Storage Oscilloscope
- Measurement of voltage, frequency, and phase
- Signal generators and analytical instruments

Text/Reference Books:

1. D. Chattopadhyay and P. C. Rakshit, "Electronics Fundamentals and Applications", New Age International Publishers
2. J. Millman, C. Halkias, and C. D. Parikh, "Integrated Electronics", McGraw-Hill Education
3. D. A. Bell, "Electronic Devices and Circuits", Oxford University Press
4. D. P. Kothari and I. J. Nagrath, "Basic Electronics", McGraw-Hill Education
5. J. D. Ryder, "Electronic Fundamentals and Applications", Prentice-Hall of India

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS2002			
Course Title	Data Structures			
Category	Programme Core			
LTP & Credits	L	T	P	Credits
	3	1	0	3
Total Contact Hours	48			
Pre-requisites	Basic Programming			

Learning Objective:

In this course, students will be taught about the significance of non-linear data structures with respect to the access and organization of data, various algorithmic approaches to write programs to solve problems in different engineering domains by using different data structures, merits and demerits of altered algorithms in terms of time-complexity.

Course Outcome:

CO1: To differentiate how the choices of data structure and algorithm methods impact the performance of a program.
CO2: To solve problems based on different data structures and also write programs.
CO3: To identify appropriate data structure and algorithmic methods in solving a problem.
CO4: To discuss the computational efficiency of the principal algorithms for sorting, searching, and hashing.
CO5: To compare and contrast the benefits of dynamic and static data structures implementations.

Course Content:

Module 1: Introduction of Data Structure [8L]

- Concepts of data structures, Abstract Data Type.
- Algorithms and programs, basic idea of pseudo-code, Properties of an algorithm.
- Algorithm efficiency and analysis, time and space analysis of algorithms – Order notations.
- Array: Different representations – row major, column major.
- Sparse matrix – its implementation and usage, Array representation of polynomials.

.

Module 2: Linear Data Structure [14L]

- Stack and its implementations (using array and linked list).
- Applications (Infix, Prefix, and Postfix with their conversions, Postfix Evaluation).
- Queues, circular queue, de-queue.
- Implementation of queue- linear and circular (using array and linked list).

- Recursion: Principles of recursion - use of stack, tail recursion. Applications - The Tower of Hanoi, Eight-queen problem.
- Linked List: Singly linked list – operations, doubly linked list – operations.
- Circular linked list – operations, Linked list representation of polynomial and applications

Module 3: Nonlinear Data Structure [18L]

- Trees: Basic terminologies, forest, tree representation (using array and linked list).
- Binary trees - binary tree traversal (pre-, in-, post- order).
- Threaded binary tree – operations.
- Binary search tree- operations (creation, insertion, deletion, searching).
- Concept of Max-Heap and Min-Heap (creation, deletion).
- Height balanced binary tree – AVL tree (insertion, deletion with examples only).
- Graph traversal and connectivity – Depth-first search (DFS), Breadth-first search (BFS) – concepts of edges used in DFS and BFS (tree-edge, back-edge, cross-edge, and forward-edge).
- Minimal spanning tree – Prim's algorithm, Kruskal's algorithm (basic idea of greedy methods).

Module 4: Searching and Sorting [8L]

- Sorting Algorithms: Bubble sort, Insertion sort, Selection sort – with the notion of complexity.
- Quick sort, Merge sort – with complexity, Radix sort – with complexity.
- Searching: Sequential search, Binary search, Interpolation Search– with complexity.
- Hashing: Hashing functions, Collision resolution techniques.

Text/Reference Books:

1. E. Horowitz, S. Sahni and S. Anderson-Freed, "Fundamentals of Data Structures of C", Universities Press.
2. S. Lipschutz, "Data Structures", Tata McGraw Hill Education (India) Private Limited.
3. A. M. Tanenbaum, "Data Structures in C", Pearson.
4. R. Thareja, "Data Structures Using C", Oxford.
5. A.K. Rath, A. K. Jagadev, "Data Structure Using C", Scitech Publications.
6. T. H. Coreman, "Introduction to Algorithms", MIT Press.

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YMT2003			
Course Title	Engineering Mathematics -II			
Category	ENGG, Minor			
LTP & Credits	L	T	P	Credits
	3	1	0	4
Total Contact Hours	48			
Pre-requisites	None			

Learning Objective:

In this course the students will learn about the Mathematics fundamental necessary to formulate, solve and analyse engineering problems.

Course Outcomes:

CO1: To use mathematical tools to evaluate multiple integrals and vector integrals.
CO2: To apply effective mathematical tools for the solutions of ordinary differential equations that model physical processes.
CO3: To understand the properties of Laplace Transform to evaluate multiple integrals and their usage
CO4: To understand the concept of Laplace transform to solve ordinary differential equations.
CO5: Apply mathematical concepts and techniques to analyze and solve engineering problems involving double and triple integration, ordinary differential equations, and Laplace transforms, demonstrating proficiency in mathematical modeling and problem-solving skills.

Course Content:

Module 1: Matrices [6L]

Linear Systems of Equations; Linear Independence; Rank of a Matrix; Determinant, Inverse of a matrix, rank-nullity theorem; System of linear equations; Symmetric, skew-symmetric and orthogonal matrices; Determinants; Eigenvalues and eigenvectors; Orthogonal transformation; Diagonalization of matrices.

Module 2: First order ordinary differential equations: [6L]

Exact, linear and Bernoulli's equations. Equations not of first degree: equations solvable for p, equations solvable for y, equations solvable for x and Clairaut's type.

Module 3: Ordinary differential equations of higher orders: [8L]

Second order linear differential equations with variable coefficients: Euler-Cauchy equations, solution by variation of parameters; Power series solutions: Legendre's equations and Legendre polynomials, Frobenius method, Bessel's equation and Bessel's functions of the first kind and their

properties.

Module 4: Complex Variable – Differentiation: [8L]

Differentiation, Cauchy-Riemann equations, analytic functions, harmonic functions, finding harmonic conjugate; elementary analytic functions (exponential, trigonometric, logarithm) and their properties; Conformal mappings, Mobius transformations and their properties.

Module 5: Complex Variable – Integration: [8L]:

Contour integrals, Cauchy-Goursat theorem (without proof), Cauchy Integral formula (without proof), Liouville's theorem and Maximum-Modulus theorem (without proof); Taylor's series, zeros of analytic functions, singularities, Laurent's series; Residues, Cauchy Residue theorem (without proof), Evaluation of definite integral involving sine and cosine, Evaluation of certain improper integrals using the Bromwich contour

Text /Reference Books:

- Erwin Kreyszig, Advanced Engineering Mathematics, 10th Edition, John Wiley & Sons, 2006.
- 2. Veerarajan T., Engineering Mathematics for first year, Tata McGraw-Hill, New Delhi, 2008. 3.
- W. E. Boyce and R. C. DiPrima, Elementary Differential Equations and Boundary Value Problems, 9th Edition, Wiley India, 2009.
- D. Poole, Linear Algebra: A Modern Introduction, 2nd Edition, Brooks/Cole, 2005.
- S. L. Ross, Differential Equations, 3rd Edition, Wiley India, 1984.
- E. A. Coddington, An Introduction to Ordinary Differential Equations, Prentice Hall India, 1995.
- E. L. Ince, Ordinary Differential Equations, Dover Publications, 1958.
- J. W. Brown and R. V. Churchill, Complex Variables and Applications, 7th Ed., Mc-Graw Hill, 2004.
- N.P. Bali and Manish Goyal, A text book of Engineering Mathematics, Laxmi Publications, Reprint, 2008.
- B.S. Grewal, Higher Engineering Mathematics, Khanna Publishers, 36th Edition, 2010.

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCH2001			
Course Title	Engineering Chemistry			
Category	Basic Science			
LTP & Credits	**L**	**T**	**P**	Credits
	3	0	0	3
Total Contact Hours	36			
Pre-requisites	None			

Learning Objective:

The objective of the Chemistry-I is to acquaint the students with the basic phenomenon/concepts of chemistry, the student faces during course of their study in the industry and Engineering field. The student with the knowledge of the basic chemistry will understand and explain scientifically the various chemistry related problems in the industry/engineering field. The student will able to understand the new developments and breakthroughs efficiently in engineering and technology. The introduction of the latest (R&D oriented) topics will make the engineering student upgraded with the new technologies.

Course Outcome:

CO1: Analyze microscopic chemistry in terms of atomic and molecular orbitals and intermolecular forces.
CO2: Rationalize bulk properties and processes using thermodynamic considerations.
CO3: Distinguish the ranges of the electromagnetic spectrum used for exciting different molecular energy levels in various spectroscopic techniques
CO4: Rationalize periodic properties such as ionization potential, electro negativity, oxidation states and electro negativity.
CO5: List major chemical reactions that are used in the synthesis of molecules.

Course Contents:

Module 1: Atomic and molecular structure [12L]

Schrodinger equation. Particle in box solutions and their applications for conjugated molecules and nanoparticles. Forms of the hydrogen atom wave functions and the plots of these functions to explore their spatial variations. Molecular orbitals of diatomic molecules and plots of the multicentre orbitals. Equations for atomic and molecular orbitals. Energy level diagrams of diatomic. Pi-molecular orbitals of butadiene and benzene and atomicity. Crystal field theory and the energy level diagrams for transition metal ions and their magnetic properties. Band structure of solids and the role of doping on band structures.

Module 2: Spectroscopic techniques and applications [6L]

Principles of spectroscopy and selection rules. Electronic spectroscopy. Fluorescence and its applications in medicine. Vibration and rotational spectroscopy of diatomic molecules. Applications. Nuclear magnetic resonance and magnetic resonance imaging, surface characterization techniques. Diffraction and scattering.

Module 3: Intermolecular forces and potential energy surfaces [4L]

Ionic, dipolar and van Der Waals interactions. Equations of state of real gases and critical phenomena. Potential energy surfaces of H3, H2F and HCN and trajectories on these surfaces.

Module 4: Use of free energy in chemical equilibrium [6L]

Thermodynamic functions: energy, entropy and free energy. Estimations of entropy and free energies. Free energy and emf. Cell potentials, the Nernst equation and applications. Acid base, oxidation reduction and solubility equilibrium. Water chemistry. Corrosion. Use of free energy considerations in metallurgy through Ellingham diagrams.

Module 5: Periodic properties [4L]

Effective nuclear charge, penetration of orbitals, variations of s, p, d and f orbital energies of atoms in the periodic table, electronic configurations, atomic and ionic sizes, ionization energies, electron affinity and electro negativity, polarizability, oxidation states, coordination numbers and geometries, hard soft acids and bases, molecular geometries

Module 6: Stereochemistry [4L]

Representations of 3 dimensional structures, structural isomers and stereoisomers, configurations and symmetry and chirality, enantiomers, diastereomers, optical activity, absolute configurations and conformational analysis. Isomerism in transitional metal compounds

.

Text /Reference Books:

1. University chemistry, by B. H. Mahan
2. Chemistry: Principles and Applications, by M. J. Sienko and R. A. Plane
3. Fundamentals of Molecular Spectroscopy, by C. N. Banwell
4. Engineering Chemistry (NPTEL Web-book), by B. L. Tembe, Kamaluddin and M. S. Krishnan

5. Physical Chemistry, by P. W. Atkins (vi) Organic Chemistry: Structure and Function by K. P. C. Volhardt and N. E. Schore, 5th Edition http://bcs.whfreeman.com/vollhardtschore5e/default.asp

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YED2001
Course Title	Professional Communication
Category	Humanities
LTP & Credits	L: 1, T: 1, P: 0, Credits: 1
Total Contact Hours	24
Pre requisites	None

Learning Objective:
In this course, the students will develop communicative competence in English so as to make them industry-ready, with special emphasis on knowledge in grammar and English writing.

Course Outcome:
CO1: To learn how to employ communication skills in the workplace
CO2: To understand and learn about the use of the different elements of English
CO3: To develop requisite skills for effective reading and comprehension of texts
CO4: To learn how to compose formal, written communication
CO5: To enhance listening skills for effective communication.

Course Content:
Module 1: Communication in a Globalized World [4L]
Introduction; meaning of communication; five stages of communication; formal and informal communication; verbal and non-verbal communication; role of body language in communication; barriers to effective communication; prejudice and lack of sensitivity in communication; gender/culture neutrality.

Module 2: Functional Grammar [8L]
Articles and prepositions. Direct and indirect verbs, subject-verb agreement. Tense and voice, sentences: simple, complex and compound; direct and indirect speech..

Module 3: Reading Comprehension [6L]
Reading purposes and skills: skimming, scanning and intensive reading, reading comprehension: fictional and non-fictional prose,one-word substitution and sentence making.

1. "Where the Mind is without Fear" by Rabindranath Tagore
2. "Out of Business" by R.K. Narayan

Module 4: Writing Skills [6L]
Paragraph, formal letters, emails, job applications, order and complaint letter, importance of punctuation in writing.

Text/Reference Books:
1. Wren and Martin (Revised by N. D. V. Prasada Rao), "High School English Grammar and Composition", S. Chand Publishing.
2. S. A. Beebe and T. P. Mottet, "Business and Professional Communication – Principles and Skills and Leadership" Pearson Education.
3. Sethi and B. Adhikari, "Business Communication" Tata McGraw-Hill.

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS2505				
Course Title	LIFE SKILLS				
Category					
LTP & Credits	L	T	P	Credits	
	0	0	1	1	
Total Contact Hours	30				
Pre-requisites	None				

Learning Objective:

Life skills are those competencies that provide the means for an individual to be resourceful and positive while taking on life's vicissitudes. Development of one's personality by being aware of the self, connecting with others, reflecting on the abstract and the concrete, leading and generating change, and staying rooted in time-tested values and principles is being aimed at. This course is designed to enhance the employability and maximize the potential of the students by introducing them to the principles that underlie personal and professional success, and help them acquire the skills needed to apply these principles in their lives and careers.

Course Outcomes:

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

- **CO1:** Define and identify different life skills required in personal and professional life.
- **CO2:** Develop an awareness of the self and apply well-defined techniques to cope with emotions and stress.
- **CO3:** Explain the basic mechanics of effective communication and demonstrate these through presentations.
- **CO4:** Take part in group discussions for goal-setting, decision-making, and problem-solving, and work effectively in teams to achieve common goals.
- **CO5:** Use appropriate thinking and problem-solving techniques to solve new problems.

Course Content:

Module 1:[6L]

Overview of Life Skills: Meaning and significance of life skills, Life skills identified by WHO: Self-awareness, Empathy, Critical thinking, Creative thinking, Decision making, Problem solving, Effective communication, Interpersonal relationships, Coping with stress, Coping with emotion.
Life Skills for Professionals: Positive thinking, Right attitude, Attention to detail, Having the big picture, Learning skills, Research skills, Perseverance, Setting goals and achieving them, Helping others, Leadership, Motivation, Self-motivation, and motivating others, Personality development, IQ, EQ, and SQ.

Module 2:[7L]

Self-awareness: Definition, need for self-awareness; Coping with stress and emotions, Human values, tools and techniques of SA: Questionnaires, journaling, reflective questions, meditation, mindfulness, psychometric tests, feedback.

Stress Management: Stress, reasons and effects, identifying stress, stress diaries, the four A's of stress management, techniques, Approaches: action-oriented, emotion-oriented, acceptance-oriented, resilience, Gratitude Training.

Coping with emotions: Identifying and managing emotions, harmful ways of dealing with emotions, PATH method and relaxation techniques.

Morals, Values, and Ethics: Integrity, Civic Virtue, Respect for others, Living peacefully, Caring, Sharing, Honesty, Courage, Valuing time, Time management, Co-operation, Commitment, Empathy, Self-confidence, Character, Spirituality, Avoiding procrastination, Sense of engineering ethics.

Module 3:[5L]

21st Century Skills: Creativity, Critical Thinking, Collaboration, Problem Solving, Decision Making, Need for Creativity in the 21st century, Imagination, Intuition, Experience, Sources of Creativity, Lateral Thinking, Myths of creativity, Critical thinking Vs Creative thinking, Functions of Left Brain & Right Brain, Convergent & Divergent Thinking, Critical reading & Multiple Intelligence.

Steps in Problem Solving: Problem Solving Techniques, Six Thinking Hats, Mind Mapping, Forced Connections, Analytical Thinking, Numeric, symbolic, and graphic reasoning. Scientific temperament and Logical thinking.

Module 4:[6L]

Group and Team Dynamics: Introduction to Groups: Composition, formation, cycle, thinking, clarifying expectations, problem solving, consensus, dynamics techniques, Group vs Team, Team dynamics, Virtual teams. Managing team performance and managing conflicts, Entrepreneurship

Case Studies: Students analyze real-world case studies to apply life skills concepts to practical problems.

Module 5:[6L]

Leadership: Leadership framework, entrepreneurial and moral leadership, vision, cultural dimensions. Growing as a leader, turnaround leadership, managing diverse stakeholders, crisis management. Types of leadership, traits, styles, VUCA leadership, levels of leadership, transactional vs. transformational leaders, leadership grid, effective leaders.

Self-Assessment Quizzes: Students complete self-assessment quizzes to reflect on their strengths, weaknesses, values, and goals.

. **Text Books**

1. BarunK.Mitra,"Personality Development & Soft Skills", Oxford Publishers, Third impression,2017.

Reference Books:

1. ShivKhera,"You Can Win", MacmillanBooks,NewYork,2003.
2. ICT Academy of Kerala,"Life Skills for Engineers ", McGraw-Hill Education (India) Private Ltd.,2016.
3. Caruso, D. R. and Salovey P, "The Emotionally Intelligent Manager: How to Develop and Use the Four Key Emotional Skills of Leadership",JohnWiley & Sons, 2004.
4. Kalyana, "Soft Skill for Managers"; First Edition;Wiley Publishing Ltd,2015.
5. Larry James,"The First Book of Life Skills"; FirstEdition,EmbassyBooks,2016.

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	-	-	-	-	-	2	-	1	2	2	1	3	-	-	-
CO2	-	-	-	-		1	-	-	3	-	-	2	-	-	-
CO3	-	-	-	-	-	1	-	-	1	3	-		-	-	-
CO4	-	-	-	-	-	-	-	-	-	3	-	1	-	-	-
CO5	-	3	2	1	-	-	-	-	-	-	-	-	-	-	-

Course Code	YED2101			
Course Title	Professional communication Laboratory			
LTP & Credits	L	T	P	Credits
	0	0	2	1
Total Contact Hours	24			
Pre-requisites	None			

Learning Objective:

In this laboratory course, students will be exposed to the need for English in the workplace and equipped with good language skills, communication skills, and soft skills.

Course Outcome:

CO1: Apply different skills of technical communication in English.

CO2: Use correct pronunciation when speaking English.

CO3: Use appropriate techniques for effective and active listening.

CO4: Learn to express ideas clearly and coherently in professional settings.

CO5: Demonstrate effective verbal and non-verbal communication skills in various professional scenarios, including presentations, meetings, and interviews

Course Content:

Suggestive List of Experiments:

1. Learn about phonetics and pronunciation guide (Introduction of phonetics and phonetic table, tongue and lip movements for vowels and consonants, monophthongs/diphthongs, voiced/unvoiced, aspirated/unaspirated, minimal pairs, syllables, stress, and intonation). [4 Days]
2. Training on listening and comprehension (Active listening and its techniques, academic listening versus business listening, listening activities: answering questions, form filling, summarizing news bulletin, presentation, video clip, lecture, story). [6 Days]
3. Training on speaking skills (Basic parameters of speaking, fluency-focused activities: JAM, conversational role plays, speaking using a picture, group discussions, and personal interviews). [6 Days]
4. Laboratory project work (Making a 5-minute animation video with voiceover OR making a 10-minute documentary film). [8 Days]

Text/ Reference Books:

1. P. Ladefoged, "A Course in Phonetics", Harcourt Brace Jovanovich College Publishers.
2. J. Sullivan, "Simply Said: Communicating Better at Work and Beyond", Wiley.
3. N. Leonardo, "Active Listening Techniques: 30 Practical Tools to Hone your Communication Skills", Rockridge Press.

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS2102			
Course Title	Data Structures Laboratory			
Category	Programme Core			
LTP & Credits	L	T	P	Credits
	0	0	3	1.5
Total Contact Hours	36			

Pre-requisites	Fundamentals of Programming

Learning Objective:

In this course, students will learn about implementing different algorithmic approaches in C programs using non-linear and linear data structures to solve problems in various engineering domains.

Course Outcome:

CO1: Choose appropriate data structures for specified problem definitions.
CO2: Compare operations such as searching, insertion, deletion, and traversing mechanisms on various data structures.
CO3: Explain practical applications of data structures.
CO4: Analyze how to store, manipulate, and arrange data efficiently.
CO5: Implement various data structures using arrays and linked lists.

Course Content:

1. Experiments on Arrays [1 day]
    - Addition and Multiplication of Arrays
    - Implementation of Sparse Matrices
2. Experiments on Abstract Data Types [2 days]
    - Implementation of stack using Array
    - Applications of stack: infix to postfix conversion, expression evaluation
3. Experiments on Linked List [2 days]
    - Implementation of linked lists and its operations: insertion, deletion, and reverse
    - Implementation of stacks and queues using linked lists
    - Polynomial addition and polynomial multiplication
4. Experiments on Searching and Sorting [2 days]
    - Searching: Linear Search, Binary Search
    - Sorting: Bubble Sort, Selection Sort, Insertion Sort, Quick Sort, Merge Sort, and Heap Sort
5. Experiments on Non-linear Data Structures [2 days]
    - Traversals of binary tree, Binary Search Tree (BST), Threaded binary tree
    - Height-balanced binary tree: AVL tree (insertion, deletion)
    - B-Trees: insertion, deletion
6. Experiments on Hashing [1 day]
    - Implementation of Hash tables and its operations: searching, inserting, and deleting, handling collisions
7. Innovative Experiments [2 days]
    - Case study of solving complex problems from various engineering domains using suitable data structures (e.g., mesh analysis in electrical circuits, event-driven simulation, etc.).

Text/Reference Books:

1. C. E. Balagurusamy, "Data Structures using C", McGraw Hill.
2. E. Horowitz, S. Sahni, and S. Anderson-freed, "Fundamentals of Data Structures of C", Universities Press.
3. A. K. Sharma, "Data Structures using C", Pearson.
4. R. Thareja, "Data Structures using C", Oxford University Press.

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS2101			
Course Title	Basic Electronics Laboratory			
Category	ENGG Skill Enhancement			
LTP & Credits	L	T	P	Credits
	0	0	3	1.5
Total Contact Hours	36			
Pre-requisites	None			

Learning Objective:

1. To prepare the students to have a basic knowledge of active and passive components.
2. To develop the ability to measure signals using a multimeter, CRO.
3. To understand the characteristics of proper biasing for BJT and FET.
4. To encourage the development of circuits using diodes, transistors, FETs, and OPAMPs.

Course Outcome:

CO1: To study and evaluate the characteristics of basic electronic components (Diode, BJT, FET).
CO2: To demonstrate the working of CRO, Function Generator, Digital Multimeter, and D.C. power supply.
CO3: To sketch the V-I characteristics of p-n junction diode, Zener diode, BJT.
CO4: To construct the rectifier circuit using diode and Inverting, Non-inverting amplifiers Circuit using Op-Amp.
CO5: To determine the characteristics parameters of Op-Amp.

Course Content:

List of Experiments:

1. Familiarization with passive and active electronic components such as Resistors, Inductors, Capacitors, Diodes, Transistors (BJT). [3 days]
2. Familiarization with measuring and testing equipment like Digital Multimeter, CRO, Signal generator, and Power Supply. [2 days]
3. Study of V-I characteristics of Junction diode. [1 day]
4. Study of V-I characteristics of Zener diode. [1 day]
5. Study of Half wave and Full wave rectifier. [2 days]
6. Study of V-I characteristics of BJT. [1 day]
7. Study of an operational amplifier chip and analyze its functionality. [2 days]
8. Design of non-inverting and inverting amplifier using operational amplifier. [2 days]
9. Design of adder, differential amplifier using operational amplifier. [2 days]
10. Verification of the truth table of basic logic gates using IC. [1 day]
11. Innovative Experiment. [1 day]

Text/Reference Books:

1. J. Millman, C. Halkias, and C. D. Parikh, "Integrated Electronics", McGraw-Hill Education.
2. D. A. Bell, "Electronic Devices and Circuits", Oxford University Press.
3. D. P. Kothari and I. J. Nagrath, "Basic Electronics", McGraw-Hill Education.
4. J. D. Ryder, "Electronic Fundamentals and Applications", Prentice-Hall of India.

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	1	1	-	-	-	1	2	2	1	2	2	-	2
CO2	3	2	1	2	2	-	-	1	2	2	1	2	3	3	3
CO3	3	3	1	1	2	-	-	1	2	1	1	2	3	3	3
CO4	3	3	3	2	2	1	1	1	3	2	2	3	3	3	3
CO5	3	2	1	1	-	-	-	-	1	1	1	2	2	-	2

Course Code	YCS2103			
Course Title	Engineering Graphics & Design Laboratory			
Category	PC			
LTP & Credits	L	T	P	Credits
	0	0	3	1.5
Total Contact Hours	36			
Pre-requisites	None/If Any			

Learning Objective:

The learning objectives of engineering graphics typically include developing skills and knowledge related to the visualization, interpretation, and communication of engineering designs.

Course Outcomes (COs):

After attending the course, students would:

CO1: Get introduced to Engineering Graphics and visual aspects of design.
CO2: Know and use common drafting tools with the knowledge of drafting standards.
CO3: Able to apply computer-aided drafting techniques to represent line, surface, or solid models in different engineering viewpoints.
CO4: Able to produce part models; carry out assembly operations and show the working procedure of a designed project work using animation.
CO5: Demonstrate proficiency in using computer-aided design (CAD) software to create and modify engineering drawings, including 2D and 3D models, with an understanding of industry standards and practices.

Course Content:

List of Drawings:

- **Traditional Engineering Graphics:** Principles of Engineering Graphics; Orthographic Projection; Descriptive Geometry; Drawing Principles; Isometric Projection; Surface Development; Perspective; Reading a Drawing; Sectional Views; Dimensioning & Tolerances; True Length, Angle; intersection, Shortest Distance.

Modules:

Module 1: Introduction to Engineering Drawing

- Principles of Engineering Graphics and their significance.
- Usage of Drawing instruments.
- Lines, lettering, Dimensioning.
- Scales – Plain, Diagonal, and Vernier Scales.

Module 2: Geometrical Construction and Curves

- Construction of polygons, Parabola, Hyperbola, Ellipse, Cycloid, Archimedean Spiral, and Involute (square & circle).

Module 3: Projection of Points, Lines, Surfaces

- Orthographic projection - 1st and 3rd angle projection.

- Projection of lines and surfaces – Hexagon.

Module 4: Projection of Solids

- Cube, Pyramid, Prism, Cylinder, Cone.

Module 5: Drawing Isometric view from Orthogonal / Sectional views of Simple Solid Objects

- Principles of Orthographic Projections.
- Conventions.
- Projections of Points and lines inclined to both planes.
- Auxiliary Views.
- Isometric Views of lines, Planes, Simple, and compound Solids.
- Conversion of Isometric Views to Orthographic Views and Vice-versa.

Module 6: Sections and Sectional Views of Right Angular Solids

- Full and half-sectional views of solids.

Module 7: Development Of Surfaces

- Prism, Cylinder, Cone.

Module 8: Computer-Aided Drafting (Using AutoCAD and/or similar softwares)

- Introduction: Cartesian and Polar coordinate system, Absolute and Relative coordinates.
- Basic editing commands: Line, Point, Trace, Rectangle, Polygon, Circle, Arc, Ellipse, Polyline.
- Editing methods.
- Basic object selection methods, Window and crossing window, Erase, Move, Copy, Offset, Fillet, Chamfer, Trim, Extend, Mirror.
- Display commands: Zoom, Pan, Redraw, Regenerate.
- Simple dimensioning and text.
- Simple exercises.

Text Books:

1. Bhatt N.D., Panchal V.M. & Ingle P.R, (2014), Engineering Drawing, Charotar Publishing House.
2. K. Venugopal, Engineering Drawing + AutoCAD, New Age International publishers.

Reference Books:

1. Pradeep Jain, Ankita Maheswari, A.P. Gautam, Engineering Graphics & Design, Khanna Publishing House.
2. Agrawal B. & Agrawal C.M. (2012), Engineering Graphics, TMH Publication.
3. Shah, M.B. & Rana B.C. (2008), Engineering Drawing and Computer Graphics, Pearson Education.
4. Narayana, K.L. & P Kannaiah (2008), Textbook on Engineering Drawing, Scitech Publishers.

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCH2101			
Course Title	Engineering Chemistry Laboratory			
Category	Basic Science			
LTP & Credits	L	T	P	Credits
	0	0	3	1.5
Total Contact Hours	36			
Pre-requisites	None			

Learning Objective:

The students will learn to estimate rate constants of reactions from concentration of reactants/products as a function of time. Measure molecular/system properties such as surface tension, viscosity, conductance of solutions, redox potentials, chloride content of water, etc. Synthesize a small drug molecule and analyze a salt sample.

Course Outcome:

CO1: To understand and apply basic chemical principles
CO2: To Perform experiments and analyze data
CO3: Apply chemical knowledge to engineering applications.
CO4: To Develop experimental techniques and problem-solving skills.
CO5: Understand the importance of chemical properties and reactions in various engineering fields.

Course Contents:

1. Determination of surface tension and viscosity
2. Thin layer chromatography
3. Ion exchange column for removal of hardness of water
4. Determination of chloride content of water
5. Colligative properties using freezing point depression
6. Determination of the rate constant of a reaction
7. Determination of cell constant and conductance of solutions
8. Potentiometry - determination of redox potentials and emfs
9. Chemical analysis of a salt
10. Lattice structures and packing of spheres
11. Models of potential energy surfaces
12. Chemical oscillations- Iodine clock reaction
13. Determination of the partition coefficient of a substance between two immiscible liquids
14. Adsorption of acetic acid by charcoal

15. Use of the capillary viscosimeters to the demonstrate of the isoelectric point as the pH of minimum viscosity for gelatin sols and/or coagulation of the white part of egg . Laboratory Outcomes

Text /Reference Books:

1. University chemistry, by B. H. Mahan
2. Chemistry: Principles and Applications, by M. J. Sienko and R. A. Plane
3. Fundamentals of Molecular Spectroscopy, by C. N. Banwell
4. Engineering Chemistry (NPTEL Web-book), by B. L. Tembe, Kamaluddin and M. S. Krishnan
5. Physical Chemistry, by P. W. Atkins (vi) Organic Chemistry: Structure and Function by K. P. C. Volhardt and N. E. Schore, 5th Edition http://bcs.whfreeman.com/vollhardtschore5e/default.asp

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Semester 3 Curriculum and Syllabus

Course Code	YCS3005			
Course Title	Probability and Statistics			
Category	ENGG, Minor			
LTP & Credits	L	T	P	Credits
	3	0	0	3
Total Contact Hours	36			
Pre-requisites	None			

Learning Objective:

In this course, students will learn about the basic knowledge of probability and statistics. At the end of the course, the students will be able to solve different real-life problems in the field of artificial intelligence, data science, etc.

Course Outcome:

CO1: To explain and demonstrate the distinctive characteristics of probability distribution.
CO2: To analyze the probability of real-world uncertain phenomena by identifying probability distribution that fits the phenomena.
CO3: To explain and demonstrate the distinctive characteristics of statistics.
CO4: To apply and analyze the uses and limitations of statistical analysis.
CO5: Apply probability and statistical concepts to solve problems in artificial intelligence, data science, and other relevant fields.

Course Content:

Module 1: Basic Probability: [12L]

Probability spaces, conditional probability, independence; Discrete random variables, Independent random variables, the multinomial distribution, Poisson approximation to the binomial distribution, infinite sequences of Bernoulli trials, sums of independent random variables; Expectation of Discrete Random Variables, Moments, Variance of a sum, Correlation coefficient, Chebyshev‘s Inequality.

Module 2: Continuous Probability Distributions: [4L]

Continuous random variables and their properties, distribution functions and densities, normal, exponential and gamma densities.

Module 3: Bivariate Distributions: [4L]

Bivariate distributions and their properties, distribution of sums and quotients, conditional densities, Bayes‘ rule.

Module 4: Basic Statistics: [8L]

Measures of Central tendency: Moments, skewness and Kurtosis - Probability distributions: Binomial, Poisson and Normal - evaluation of statistical parameters for these three distributions, Correlation and regression – Rank correlation.

Module 5: Applied Statistics: **[8L]**

Curve fitting by the method of least squares- fitting of straight lines, second degree parabolas and more general curves. Test of significance: Large sample test for single proportion, difference of proportions, single mean, difference of means, and difference of standard deviations.

Text/Reference Books:

1. E.Kreyszig, "Advanced Engineering Mathematics", John Wiley & Sons.
2. B.S.Grewal, "Higher Engineering Mathematics", Khanna Publishers.
3. N.G.Das, "Statistical Methods (Combined Volume)", Tata-McGrawHill.
4. R.GargandC.Prasad, "Advanced Engineering Mathematics", Khanna Publishers.
5. S.Ross, "A First Course in Probability", Pearson Education India.
6. W.Feller, "An Introduction to Probability Theory and its Applications, Vol.1", Wiley.
7. J.E.FreundandR.E.Walpole, "Mathematical Statistics", Prentice Hall.

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3

Course Code	YCS3001			
Course Title	Digital Logic and Electronics			
Category	ProfessionalCore			
LTP & Credits	L	T	P	Credits
	3	0	0	3
TotalContactHours	36			
Pre-requisites	None			

Learning Objective:
In this course, students will learn about the representation of numbers in a computer system and how digital circuits can be designed using logic gates and flip-flops. They will also learn about the process of digital-to-analog and analog-to-digital conversions. After completing this course, students will be in a better position to understand the basic operation of a computer system and how various functional blocks can be implemented.

Course Outcome:

CO1: Explain the binary number system and its importance in digital circuit design.

CO2: Classify and analyze various ways of minimizing switching functions.
CO3: Understand the process of designing combinational logic circuits.
CO4: Understand the process of designing sequential logic circuit modules.
CO5: Understand and remember the process of analog-to-digital and digital-to-analog conversion.

Course Content:

Module 1: Number Systems and Binary Codes [5L]

- Introduction to number systems: decimal, binary, octal, hexadecimal.
- Conversion from one number system to another.
- Signed number representation: sign-magnitude, 1's complement, and 2's complement.
- Addition and subtraction of numbers.
- Binary codes: BCD, excess-3 code, gray code.
- Introduction to floating-point representation and its importance in representing real numbers in digital systems.

Module 2: Logic Families and Minimization of Switching Functions [6L]

- Logic gates and their functionalities.
- Logic families: TTL, nMOS, CMOS, pass transistor logic.
- Realization of gates.
- Boolean algebra, truth tables, and switching functions.
- Minimization of completely and incompletely specified switching functions: Karnaugh Map and Quine-McCluskey methods.
- Explanation of buffer and tri-state buffer operations and their applications in digital circuits.
- Introduction to HDL and its importance in describing digital circuits for synthesis and simulation.

Module 3: Combinational Logic Circuits [7L]

- Realization of Boolean functions using NAND/NOR gates.
- Half-adder, full-adder, and ripple-carry adder/subtractor.
- Decoders, Encoders, and Multiplexers: applications in logic design.
- Adder-Subtractor Circuits: Detailed explanation of adder-subtractor circuits and their applications in arithmetic operations.
- Comparator Circuits: Introduction to comparator circuits and their use in comparing binary numbers.

Module 4: Sequential Logic Circuits [7L]

- Clocks, flip-flops, and latches.
- Types of flip-flops: SR, D, JK, T; Edge-triggered and master-slave flip-flops.
- State table and state diagram, state minimization, synthesis of finite state machines (FSMs).

Module 5: Counters and Registers [6L]

- Synchronous and asynchronous counters, up/down counters.
- Applications of counters.
- Registers: parallel-in parallel-out and shift registers, linear feedback shift register (LFSR).
- Applications of registers in datapaths.

Module 6: D/A and A/D Conversion Techniques [5L]

- Digital-to-analog converters: principle of operation, weighted resistor and resistive ladder D/A converters.
- Analog-to-digital converters: resolution and accuracy.
- Types of A/D converters: flash type, counter type, successive-approximation type.
- Sigma-Delta ADCs: Introduction to sigma-delta analog-to-digital converters and their advantages in achieving high resolution.
- Successive Approximation Register (SAR) ADCs: Detailed explanation of successive approximation register ADCs and their operation principles.

Text/Reference Books:

1. Z. Kohavi and N. K. Jha, "Switching and Finite Automata Theory (3rd Ed.)", Cambridge University Press.
2. M. Morris Mano, "Digital Design (3rd Ed.)", Pearson.
3. G. De Micheli, "Synthesis and Optimization of Digital Circuits", Tata-McGraw-Hill

CO-POMapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS3002			
Course Title	Computer Organization and Architecture			
Category	ENGG, Major			
LTP & Credits	L	T	P	Credits
	3	0	0	3
Total Contact Hours	36			
Pre-requisites	a)Digital Circuits and Logic Design			

Learning Objective:

In this course, students will learn about the evolution of computer systems and development in computer organization and architecture. Special emphasis will be placed on the various functional units of a computer system and how instructions are executed. By the end of the course, students will have a better understanding of how programs are executed in a computer system.

Course Outcomes:

CO1: Explain the process of instruction execution.

CO2: Analyze and design the control unit of a computer system.

CO3: Analyze and design adder, multiplier, and division units.

CO4: Analyze and design memory subsystems.

CO5: Explain and classify various input/output data transfer techniques.

Course Content:

Module 1: Evolution of Computer System [4L]

- Introduction to computing system
- Computer organization and architecture
- Basic functional units of a computer
- Evolution of computers
- Stored-program concept
- Von Neumann and Harvard models

Module 2: Basic Operation of Computer [7L]

- Instruction Set Architecture
- CPU registers
- Instruction format and encoding
- Addressing modes
- Instruction set
- Instruction types
- Instruction decoding and execution
- Basic instruction cycle
- Reduced Instruction Set Computer (RISC)
- Complex Instruction Set Computer (CISC)
- Case study: MIPS Instruction set, MIPS assembly language programming
- Pipelining: Basic concepts, pipeline hazards, techniques to overcome hazards (e.g., forwarding, stalling, branch prediction)
- Case study: Pipeline stages and operation in a modern processor (e.g., Intel or AMD processors)

Module 3: Processor Unit Design [7L]

- Register transfer operations

- Internal single and multi-bus architecture
- Design of control unit: hardwired control unit design, micro programmed control unit design, concept of control word and control store
- Horizontal, vertical, and diagonal micro programmed control unit design

Module 4: Arithmetic Unit Design [6L]

- Adder and subtractor
- Shift-and-add multiplication
- Signed multiplication: Booth's algorithm, integer division, restoring and non-restoring division
- Floating-point representation: IEEE floating point format, floating-point arithmetic
- Superscalar and out-of-order execution: Concepts and advantages
- Speculative execution: Purpose and implementation in modern processors

Module 5: Memory Unit Design [7L]

- Basic memory types: Random Access Memory (RAM), Read Only Memory (ROM), Static RAM, Dynamic RAM
- Memory hierarchy
- Cache memory: mapping techniques, memory interleaving
- Virtual memory: Concepts, benefits, implementation (e.g., page tables, TLB)
- Memory management techniques: Fragmentation, compaction, techniques to reduce fragmentation (e.g., buddy system)

Module 6: Input Output Organization [5L]

- I/O mapped I/O and Memory mapped I/O
- Synchronous and Asynchronous serial data communication
- Secondary memory: disk, flash memory
- I/O Data transfer techniques: Programmed I/O, Interrupt-driven I/O, Direct Memory Access (DMA)

Text/Reference Books:

1. C. Hamacher, Z. Vranesic, and S. Zaky, “Computer Organization (5th Ed.)”, Tata-McGraw-Hill.

2. W. Stallings, “Computer Organization and Architecture (6th Ed.)”, Prentice Hall of India.

3. D. A. Patterson, and J. L. Hennessy, “Computer Organization and Design – The Hardware/Software Interface”, Morgan Kaufmann.

CO-POMapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS3003			
Course Title	Design and Analysis of Algorithms			
Category	ProfessionalCore			
LTP & Credits	L	T	P	Credits
	3	0	0	3
TotalContactHours	36			

Learning Objective: The course will cover topics such as algorithm complexity concepts and diverse algorithmic designs, including divide and conquer, dynamic programming, and greedy algorithms. It will also include important search and sorting algorithms, graphs, and basic approaches to optimization.

Course Outcome:

1. To understand the concepts of time and space complexity, worst case, average case, and best case complexities, and the big-O notation.
2. To apply design principles and concepts to algorithm design.
3. To understand and analyze the mathematical foundation in the analysis of algorithms.
4. To explain and classify different algorithmic design strategies.
5. To analyze the efficiency of algorithms using time and space complexity theory.

Course Content:

Module 1: Complexity Analysis [4L]

- Time and space complexity
- Different asymptotic notations and their mathematical significance
- Solving recurrences: substitution method, recurrence tree method, Master Theorem

Module 2 Divide and Conquer [8L]

- Basic concept
- Examples: binary search, merge sort, quick sort and their complexity (all three cases)
- Heap sort and its complexity
- Karatsuba algorithm
- Introduction to randomized algorithms and their application in divide and conquer
- Lower Bound Theory: Comparison trees, Oracle and adversary argument, State space method

Module 3: Dynamic Programming [10L]

- Basic concepts
- Matrix chain manipulation
- Strassen's algorithm
- Longest common subsequence
- All-pair shortest paths (Floyd Warshall)
- Single-source shortest path (Dijkstra, Bellman-Ford)
- 0/1 Knapsack problem
- Traveling Salesman problem
- Greedy Method: Basic concept
- Examples: fractional Knapsack problem, job sequencing with deadlines
- Minimum cost spanning tree using Prim's and Kruskal's method
- Huffman encoding and decoding
- Advanced dynamic programming techniques (e.g., memoization, tabulation)
- Backtracking: Basic concept
- Examples: n-queens problem, graph coloring problem
- Disjoint Set Manipulation: Set manipulation algorithm like UNION-FIND, union by rank

Module 4: String Matching Problem [8L]

- Different techniques: Naive algorithm, string matching using finite automata, and Knuth, Morris, Pratt (KMP) algorithm with their complexities
- Rabin-Karp algorithm for string matching with a rolling hash function
- Boyer-Moore algorithm for string searching with a heuristic to skip comparisons
- Amortized Analysis and Network Flow: Aggregate, Accounting, and Potential Method
- Ford Fulkerson algorithm
- Max-Flow Min-Cut

Module 5: Notion of NP-Completeness [6L]

- NP class, NP-hard class, NP-complete class, their interrelationship
- Reductions and Polynomial time Reducibility
- Satisfiability problem (3-SAT and 2-SAT)
- Cook-Levin's theorem
- Clique decision problem
- Vertex Cover problem
- Additional small important topics in existing modules

Text/Reference Books:

1. T.H. Cormen, C.E. Leiserson, R.L. Rivest, and C. Stein, "Introduction to Algorithms", MIT Press.
2. E. Horowitz and S. Sahni, "Fundamentals of Computer Algorithms", Universities Press.
3. K. Mehlhorn and P. Sanders, "Data Structures and Algorithms", Springer.
4. A. Aho, J. Hopcroft, and J. Ullman, "Design and Analysis of Computer Algorithms", Addison-Wesley.
5. D.E. Knuth, "The Art of Computer Programming (Vol.3)", Addison-Wesley.

.

CO-PO Mapping:

Course Outcome (CO)	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO-1**	**PSO-2**	**PSO-3**

Course Code	YCS3004
Course Title	Computational Biology

CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Category	SCI			
LTP & Credits	L	T	P	Credits
	3	0	0	3
Total Contact Hours	36			
Pre-requisites	None/ If Any			

Learning Objective:

The course is explained the principles of systems biology and the central dogma of molecular biology. Explore various biological databases and understand their structure, including sequence and gene expression data.This course aanalyze biological sequences through alignment and similarity measures by applying dynamic programming and heuristic algorithms for sequence alignment. Use Hidden Markov Models (HMM) for pair wise alignment and motif discovery

Course Outcome:

CO1: Explain key concepts of cell biology, systems biology, and the central dogma, along with the role of biological databases in bioinformatics.
CO2: Retrieve and interpret biological sequence and expression data from major public databases such as NCBI and assess the challenges of biological data integration.
CO3: Apply sequence alignment techniques including dynamic programming, heuristic methods, and Hidden Markov Models to analyze genetic sequences.
CO4: Construct and evaluate phylogenetic trees and perform motif discovery using computational methods.
CO5: Implement clustering and classification algorithms to analyze biological data, and interpret the results in the context of biological information systems.

Course Content:

Module 1: Cell Biology [10L]

System Biology, Central Dogma, Biological Database: Sequence data, Gene expression data, Micro-array experiment, NCBI database, Challenges faced in the integration of biological data, Data management and data integration in bio-informatics, Issues related to the designing of a biological information system.

Module2: Sequence similarity, homology, and alignment. Pair wise alignment [14L]

Scoring model, dynamic Programming algorithms, heuristic alignment, and pair wise alignment using Hidden Markov models (HMM), multiple alignment**:** scoring model, local alignment gapped and ungapped global alignment. Motif finding: motif models, finding occurrence of known sites, discovering new sites, Amino Acid, Protein, Phylogenetic tree construction: Neighbor Joining Algorithm.

Module 3:Clustering algorithms: [12]

K-means, k-medoid, Isodata, AGNES, DIANA, BIRCH, DBSCAN, CHAMELON, Grid based methods, Model based methods Classifier: Bayes theorem, Naïve Bayes classifier, Bayesian belief network,

Text /Reference Books:

1. Molecular Biology of the Cell, Bruce Alberts, Alexander Johnson, Julian Lewis, Martin Raff, Keith Roberts, Peter Walter, New York: Garland Science; 2002
2. Bioinformatics: Sequence and Genome Analysis, David W. Mount, Cold Spring Harbor Laboratory Press
3. Bioinformatics And Functional Genomics: A Short Course, Jonathan Pevsner, Wiley-Liss
4. Data Mining Concepts and tecniques, Han &Kamber,Elsevier
5. Introduction to Bioinformatics, Arthur M. Lesk, Oxford University Press
6. Developing Bioinformatics Computer Skills,CynthiaGibas, Per Jambeck,O'Reilly
7. An Introduction to Bioinformatics Algorithms,Neil C. Jones, Pavel A. Pevzner,MIT Press (MA).

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	1	1	-	-	1	1	-	-	2	-	1	1			
CO2	2	1	2	-	1	1	-	-	2	-	1	2			
CO3	2	2	-	-	1	1	-	-	2	-	1	2	2		
CO4	2	2	1	-	1	1	-	-	2	-	1	2		1	
CO5	2	3	2	-	1	1	-	-	2	-	1	2	2		

CourseCode	YCS3101			
CourseTitle	Digital Logic and Electronics Laboratory			
Category	ProfessionalCore			
LTP&Credits	L	T	P	Credits
	0	0	3	1.5
TotalContactHours	36			
Pre-requisites	None			

Learning Objective:

In this laboratory course, students will conduct hands-on sessions for the design and implementation of combinational and sequential digital circuit modules, as well as interfacing LED and 7-segment display units.

Course Outcome:

CO1: To understand and test the functionalities of basic gates.
CO2: To understand Boolean functions using various combinational circuit modules (like gates, multiplexer, decoder, etc.).
CO3: To understand and verify the functions of flip-flops and other sequential circuit elements (like counter, register, etc.).
CO4: To understand and analyze complex digital systems and verify the functionality.
CO5: To design a data path consisting of an ALU, registers, and multiplexers, and then design the control path to compute the Greatest Common Divisor (GCD) of two numbers.

Course Content:

1. Design a basic inverter using transistors, obtain the transfer characteristics, and measure the propagation delay. Repeat the experiment using an inverter chip.
2. Given a Boolean function, minimize it and realize the function using NAND gates. Using a 555 timer, design a rectangular waveform generator of a given frequency.
3. Design a full-adder using basic gates. Cascade two such full-adders to realize a 2-bit adder. Connect LEDs to observe the outputs and verify the functionality.
4. Verify the functionality of multiplexer and decoder chips. Implement a 4-variable Boolean function using an 8-to-1 multiplexer.
5. Implement RS and JK master-slave flip-flops using NAND gates and verify their functionalities. Verify the functionality of a J-K flip-flop chip.
6. Using JK or D flip-flops, design a 4-bit shift register and verify the functionality. Modify the designs to make it into (a) a ring counter, (b) Johnson counter and verify the functionality.
7. Design a 3-bit synchronous counter that counts in some arbitrary count sequence. Apply a square wave at the clock input and analyze the waveforms observed.
8. Design a 2-digit BCD counter and display the count value on 7-segment display units.
9. Design an 8-bit modulo-N counter for some arbitrary value of N. Connect a D/A converter at the output of the counter and observe the output waveform. Analyze the operation for various values of N. Use the circuit to display the transfer characteristic of a NOT gate on the oscilloscope.
10. Design a data path consisting of an ALU, registers, and multiplexers. Hence design the control path to compute the GCD of two numbers.

Text/Reference Books:

1. Z. Kohavi and N. K. Jha, “Switching and Finite Automata Theory (3rd Ed.)”, Cambridge University Press.
2. M. Morris Mano, “Digital Design (3rd Ed.)”.
3. G. De Micheli, “Synthesis and Optimization of Digital Circuits”, Tata-McGraw-Hill.

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

CourseCode	YCS3104			
CourseTitle	IT Workshop Lab(SciLab/MATLAB/C++)			
Category	ENGG, Skill Enhancement Course			
LTP&Credits	L	T	P	Credits
	0	0	3	3
TotalContactHours	36			

Pre-requisites	None

Learning Objective:

The purpose of this course is to learn and practice basic programming concepts, skills needed for basic problem solving, and applications—plots and built-in statistical and set operations, all using MATLAB as the vehicle. MATLAB is a powerful software package that has built-in functions to accomplish a diverse range of tasks, from mathematical operations to three-dimensional imaging. Additionally, MATLAB has a complete set of programming constructs that allow users to customize programs to their own specifications.

Course Outcome:

CO1: To impart knowledge to the students using MATLAB software that enhances programming knowledge in Research and Development.

CO2: To introduce the MATLAB technical computing environment for the analysis and visualization of data to understand and predict future scenarios.

CO3: To introduce students to the use of a high-level programming language, MATLAB, for solving scientific and engineering problems.

CO4: Reinforce a structured, top-down approach to formulate and solve problems.

CO5: Introduce common approaches, structures, and conventions for creating and evaluating computer programs, primarily in a procedural paradigm.

Course Content:

1. **Basic Commands:**
    - Clear Command: Removes all variables from the workspace.
    - Clc Command: Clears the command window and homes the cursor.
    - Help Command: Displays help about a topic if it exists.
    - Lookfor Command: Provides help by searching through all the first lines of MATLAB help topics and returning those that contain a keyword you specify.
    - Edit Command: Enables you to edit (open) any M-file in the Editor Window. This command does not open built-in functions like sqrt. See also the type Command.
    - More command: More on enables paging of the output in the MATLAB command window, and more off disables paging of the output in the MATLAB command window.
2. **Create/Initialize Matrix:**
    - Separate the elements of a row with blanks or commas, use semicolons to indicate the end of each row, and surround the entire list of elements with square brackets.
3. **Matrix Operations:**
    - Display the result of addition, subtraction, and multiplication operations on matrices.
4. **Colon Operator:**
    - Create and display vectors using the colon operator along with spacing, perform possible arithmetic operations, and validate the result.
5. **Zero and Unit Matrix:**

- Create and display zero and unit matrices of the required size.

6. **Matrix Manipulations:**
    - Find and display the inverse and transpose of a given matrix, find the largest element in each vector of the matrix, and store it at n+1 positions of each row or vector. Also, find the maximum element of the matrix at n+1, m+1 positions, where n and m are the sizes of the given matrix.
7. **Display Results of Various Commands/Functions:**
    - Diag(), prod(), median(), sort(), det(), tril(), and triu().
8. **Built-in Functions:**
    - Write a program to display the result of various built-in functions like abs(), atan(), ceil(), log(), rem(), sqrt(), rand(), randn(), and mod().
9. **Plotting:**
    - Plot a graph of y versus x, where x values range from zero to 2*pi, and y values are the sine of x.
10. **Graph Labeling:**
    - Display the graph with proper labeling using xlabel, ylabel, and title commands.
11. **Plotting Sinc Function:**
    - Plot the Sinc function, where Sinc(x) = sin(x) / x, and $-2\pi \leq x \leq 2\pi$.
12. **M-File Program:**
    - Write a program in an M-File to read a 3x3 Matrix, then display the diagonal of the matrix.
13. **String Manipulation:**
    - Write a program to read a string, then replace each character in the string with its following character in ASCII code.
14. **Control Structures:**
    - Programs using control structure statements. For example, display all prime numbers between two given input values.
15. **Image Processing:**
    - Write a program to read an image and display the resulting image of the following commands: imresize() and imhist().

Text/Reference Books:

1. Rafael C. Gonzalez, Richard E. Woods, Steven L. Eddins, Digital Image Processing MATLAB, Pearson Education Inc., 2004.
2. Stormy Attaway, MATLAB: A Practical Introduction to Programming and Problem Solving, Butterworth-Heinemann Publisher, 2013.
3. MathWorks Book by Cleve Moler
4. MATLAB Getting Started Guide

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS3102			
Course Title	Computer Organization and Architecture Laboratory			
Category	ENGG, Major			
LTP & Credits	L	T	P	Credits
	0	0	3	1.5
Total Contact Hours	36			
Pre-requisites	a)Digital Circuits Laboratory			

Learning Objective:

In this laboratory course, students will conduct experiments using a MIPS instruction set simulator. They will also learn how to model various hardware blocks using the hardware description language Verilog. They shall be designing various functional units like adder, multiplier, processor, etc., using Verilog.

Course Outcome:

1. **CO1: To understand how to write assembly language programs in MIPS.**
2. **CO2: To design various combinational and sequential circuits using Verilog.**
3. **CO3: To design and analyze various CPU functional units using Verilog.**
4. **CO4: To apply a pipelined processor using Verilog.**
5. **CO5: To implement and simulate the MIPS 5-stage pipeline in Verilog, utilizing a subset of 16 instructions, and validate the design by composing a comprehensive test bench with sample machine language programs stored in a memory module.**

Course Content:

1. **Familiarization with MIPS assembly language programming using an instruction set simulator like QtSPIM.**
    - **Reading and displaying an arbitrary string and an integer.**
    - **Storing numbers sequentially in memory and finding the minimum, maximum, and sum.**
    - **Sorting a set of numbers stored in memory. [2 days]**
2. **Familiarization of function calls with MIPS assembly language programming.**
    - **Writing a function to compute the factorial of a given number.**
    - **Writing a function to compute the GCD of two numbers.**
    - **Writing a function to compute the N-th Fibonacci number. [2 days]**
3. **Familiarization with a Verilog simulator like iVerilog and writing simple combinational and sequential modules using behavioral and structural modeling with Verilog.**
    - **Writing a module to implement an arbitrary Boolean function (e.g., F=A'BC+C'D).**
    - **Writing a module to implement a full adder and hence a 4-bit ripple carry adder.**
    - **Writing a module to implement a D flip-flop and hence a 4-bit shift register.**
    - **Writing a module to implement an 8-bit up-down counter with asynchronous clear. [2 days]**
4. **Writing Verilog modules to implement functional blocks used in computer organization.**
    - **Writing a module to implement a 16-bit arithmetic and logic unit with 8 functions.**
    - **Writing a module to implement read/write operations in a 1024x16 memory system. [2 days]**
5. **Implementing the MIPS 5-stage pipeline in Verilog, using a subset of 16 instructions. The design has to be tested by writing a test bench containing sample machine language programs stored in a memory module. [4 days]**

Text/Reference Books:

1. **qtSPIM simulator, http://spimsimulator.sourceforge.net/**
2. **MIPS overview, https://tams.informatik.uni-hamburg.de/applets/hades/webdemos/mips.html**
3. **M. M. Mano and M. D. Ciletti, "Digital Design: with an Introduction to Verilog HDL (5th Ed.)", Pearson Education.**
4. **J. Bhasker, "Verilog HDL Synthesis: A Practical Primer", B.S. Publications.**

CO-PO Mapping:

Course Outcome (CO)	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

CourseCode	YCS3103			
CourseTitle	Design and Analysis of Algorithms Laboratory			
Category	ProfessionalCore			
LTP&Credits	L	T	P	Credits
	0	0	3	1.5
TotalContactHours	36			
Pre-requisites	a)ProgrammingPractices I			

Learning Objective:

The course aims to provide strategies (divide and conquer, dynamic programming, greedy method) to solve problems in computer science effectively. Using these paradigms, the course will demonstrate innovative and effective approaches to solving specific problems. Rigorous proof of algorithm validity will be emphasized.

Course Outcome:

CO1: To prove the correctness and analyze the running time of basic algorithms.

CO2: To design algorithms using dynamic programming, greedy method, backtracking, branch and bound strategy, and to describe algorithms that employ these strategies.

CO3: To compare, contrast, and choose appropriate algorithmic design techniques to present an algorithm that solves a given problem.

CO4:To identify and analyze criteria and specifications appropriate to new problems.

CO5:To apply algorithmic design techniques to solve real-world scheduling and routing problems, demonstrating proficiency in graph coloring, backtracking, and optimization approaches.

Course Content:

1. Experiments on Divide and Conquer Approach:
    - Binary Search (Recursive & Iterative)
    - Merge Sort, Heap Sort, Quick Sort
    - Find Maximum and Minimum element from an array of integers Duration: 2 days
2. Experiments on Dynamic Programming:
    - Minimum number of scalar multiplications needed for chain of matrix
    - All pair of shortest paths for a graph
    - Single-source shortest path for a graph (Dijkstra, Bellman-Ford)
    - Longest common subsequence problem Duration: 2 days
3. Experiments on Backtracking:
    - The n-Queens problem

- Graph Coloring problem Duration: 2 days

4. Experiments on Greedy Methods:
    - Knapsack problem
    - Job sequencing with deadlines
    - Minimum cost spanning tree by Prim's and Kruskal's algorithm Duration: 2 days
5. Innovative Experiments:
    - Take the university timetable for all departments. Write a computer program to find all conflicts within the timetable using graph coloring approach. Provide a solution using Backtracking. Compute the distance and find the stoppages every classmate of yours cover to reach the institute. Then assume their speeds based on their travelling modes. Compute each student's minimum time to reach the institute premises. Duration: 2 days

Text/Reference Books:

1. T. H. Cormen, C. E. Leiserson, R. L. Rivest, and C. Stein, "Introduction to Algorithms", MIT Press.
2. E. Horowitz and S. Shani, "Fundamentals of Computer Algorithms", Universities Press.
3. K. Mehlhorn and P. Sanders, "Data Structures and Algorithms", Springer.

.

CO-POMapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

UNIVERSITY

Semester 4 Curriculum and Syllabus

Course Code	YCS4005			
Course Title	Discrete Mathematics			
Category	ENGG , Minor			
LTP & Credits	L	T	P	Credits
	3	0	0	3
Total Contact Hours	36			
Pre-requisites	None			

Learning Objective:

In this course, the students will learn about the mathematical foundations of computer science. The specific topics that would be covered include propositional calculus and proof techniques, set theory and other derived algebraic structures, recurrence relations, and the theory of graphs. The course will be very helpful for the students as it acts as a prerequisite for various next-level courses like algorithms, automata theory, artificial intelligence, etc.

Course Outcome:

CO1: To explain the distinctive characteristics of propositional logic and its applications.
CO2: To demonstrate the applications of various proof techniques.
CO3: To explain the basic concepts of sets, relations, functions, and various algebraic structures.
CO4: To understand the concept of recurrence relations and methods of solution.
CO5: To explain and analyze the concept of graphs and various graph algorithms.

Course Content:

Module 1: Propositional Logic [6L]

- Introduction to Propositional Calculus: Propositions, Logical Connectives, Conjunc- tion, Disjunction, Negation.
- Conditional Connectives, Implication, Converse, Contra- positive, Inverse, Bi-conditional statements, Logical Equivalence, Tautology.
- Conjunctive and disjunctive normal forms.
- Universal and existential quantifiers and their negations.

Module 2: Proof Techniques [6L]

- Forward proof, proof by contradiction, contrapositive proofs, proof by mathematical induction, proof of necessity and sufficiency.
- Weak Induction and Strong Induction.

Module 3: Sets, Relations and Functions [8L]

- Operations on sets, relations and functions, binary relations, partial ordering relations, equivalence relations.
- Finite and infinite set, countable and uncountable sets.
- Algebraic structures with one binary operation: semigroups, monoids, and groups.
- Algebraic structures with two binary operations: rings and fields.
- Power Sets, Cardinality of Finite and Infinite Sets.

Module 4: Recurrence Relations [6L]

- Recurrence relations: Formulation of different counting problems in terms of recurrence relations.

- Solution of recurrence relations with constant coefficients by (i) iterative method, (ii) characteristic roots method, (iii) generating functions method.
- Pigeonhole principle.

Module 5: Introduction to Graphs [10L]

- Graphs and their basic properties: digraphs, weighted graph, connected and disconnected graph, bipartite graph, complement of a graph, regular graph, complete graph, walk, path, circuit, Euler graph, cutset, cutvertices, adjacency and incidence matrices of a graph, isomorphism.
- Graph coloring problem, planar graphs, trees.
- Hamiltonian Graphs, Dijkstra's Algorithm.

Text/Reference Books:

1. C.L. Liu, "Elements of Discrete Mathematics", Tata McGraw-Hill.
2. J-P. Tremblay and R. Manohar, "Discrete Mathematical Structures with Applications to Computer Science", McGraw-Hill Education.
3. S.K. Chakraborty and B.K. Sarkar, "Discrete Mathematics", Oxford University Press.
4. R. Graham, D.E. Knuth, and O. Patashnik, "Concrete Mathematics: A Foundation for Computer Science", Addison-Wesley.
5. N. Deo, "Graph Theory with Applications to Engineering and Computer Science", Prentice-Hall.

CO-PO Mapping:

Course Outcome (CO)	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO-1**	**PSO-2**	**PSO-3**
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS4001			
Course Title	Operating Systems			
Category	ENGG, Major			
LTP & Credits	L	T	P	Credits
	3	0	0	3
Total Contact Hours	36			
Pre-requisites	Data Structures and Algorithms Computer Organization and Architecture			

Learning Objective:

In this course, students will learn about the role of an operating system as the interface between application programs and computer hardware. They will understand how the operating system manages various computer resources, which is essential for handling large software projects effectively.

Course Outcomes:

CO1: To explain the role of an operating system and how it acts as an interface between hardware and software.
CO2: To contrast the concepts of processes and threads, and how they are scheduled.
CO3: To demonstrate the use of various synchronization tools in solving the critical section problem.
CO4: To explain and classify various memory management techniques, including virtual memory.
CO5: To apply the knowledge of data structures to explain how file systems can be implemented on secondary storage.

Course Content:

Module 1: Introduction to Operating Systems [4L]

- Functionalities of an operating system – hardware/software interface.
- Evolution of operating systems – batch, multi-programmed, time-sharing, real-time, distributed.
- Simultaneous Peripheral Operations On-Line (SPOOL).
- Protection and Security – user/supervisory mode, privileged instructions, system calls (invoking OS services).

Module 2: Processes and Threads [7L]

- Processes – basic concept, process control block (PCB), process state transition diagram.
- Process scheduling – independent and cooperating processes, inter-process communication using shared memory and message passing. Case studies from Unix/Linux.
- Threads – lightweight process concept, benefits of threads, user and kernel-level threads, using thread library in Unix/Linux.
- CPU Scheduling – scheduling criteria, preemptive and non-preemptive scheduling. Scheduling algorithms – FCFS, SJF, SRTF, RR, priority, multi-level feedback queue.

Module 3: Process Synchronization and Deadlocks [7L]

- Classical problems of process synchronization – producer-consumer, reader-writer, dining philosopher, etc.
- Critical section problem – illustration, software solutions, solution using synchronization hardware: test-and-set (TST) and SWAP instructions.
- Semaphores – definition, binary and counting semaphores, implementation of semaphores, minimizing busy waiting. Case studies from Unix/Linux.
- Deadlocks – deadlock characterization, methods of handling deadlock, deadlock prevention versus deadlock avoidance, Banker's algorithm.

Module 4: Memory Management [8L]

- Logical versus physical address space, swapping, contiguous memory allocation, memory protection using fence registers.
- Paging – basic concept, performance analysis, translation look-aside buffer (TLB). Segmentation.
- Virtual memory – separation of logical and physical address space, demand paging, locality of reference. Page replacement algorithms – FCFS, LRU, Optimal, Belady's anomaly. Thrashing, working set model.
- Page table structure and its optimization techniques. Segmentation with paging (combined scheme). Memory protection and access control mechanisms. Shared memory and memory-mapped files.
- Case studies on memory management in modern operating systems (e.g., Windows, macOS, Linux).

Module 5: Device and File Management [7L]

- Disk structure – cylinders, tracks, and sectors.
- Disk scheduling algorithms – FCFS, SSTF, SCAN, C-SCAN, LOOK, C-LOOK.
- File system – file concept, access methods, directory and file system structure, allocation methods (contiguous, linked, indexed), free space management. Case study for Unix/Linux.
- Disk management optimizations (e.g., track skewing, clustering, buffering), RAID levels and their implications for performance and fault tolerance.
- File system reliability and recovery mechanisms (e.g., journaling, shadowing), Network file systems and distributed file systems (e.g., NFS, AFS, CIFS).
- Case studies on file systems used in cloud computing and distributed systems.

Module 6: Miscellaneous Topics **[3L]**

- Brief overview of real-time and distributed operating systems, mobile operating systems.

Text/Reference Books:

1. A. Silberschatz, P. B. Galvin, and G. Gagne, "Operating System Concepts", Wiley Asia.
2. D. M. Dhamdhere, "Operating Systems: A Concept-Based Approach", Tata McGraw-Hill.
3. M. Bach, "Design of the Unix Operating System", Prentice-Hall of India.
4. W. Stallings, "Operating Systems: Internals and Design Principles", Prentice-Hall of India.
5. C. Crowley, "Operating System: A Design-Oriented Approach", Irwin Publishing.
6. G. J. Nutt, "Operating Systems: A Modern Perspective", Addison-Wesley.

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS4002			
Course Title	Formal Language and Automata Theory			
Category	ENGG, Major			
LTP & Credits	L	T	P	Credits
	3	0	0	3
Total Contact Hours	36			
Pre-requisites	a) DiscreteMathematics b) ProgrammingandDataStructure			

Learning Objective:

In this course, students will learn the theory of computation, including formal language classes and their relationships, techniques to prove or disprove theorems in automata theory using its properties, and approaches to determine the decidability and intractability of computational problems. At the end of the course, students will be able to analyze complex problems and automata to find solutions.

Course Outcome:

CO1: Explain the basic properties of formal languages and grammars.

CO2: Understand the tools for recognizing different formal languages.

CO3: Differentiate between regular, context-free, and recursively enumerable languages.

CO4: Apply the theory of computation and computational models, including decidability and intractability.

CO5: Analyze complex problems and automata to find solutions using the theory of computation and computational models, including decidability and intractability.

Course Content:

Module 1: Introduction to Finite Automata [10L]

- Finite Automata, Alphabets, Strings, Languages
- Regular Languages, Deterministic Finite Automata (DFA)
- Nondeterministic Finite Automata (NFA)
- Equivalence of NFA and DFA
- Minimization of Finite Automata
- Myhill-Nerode Theorem
- FA with output: Moore and Mealy machine

Module 2: Properties of Regular Expression [7L]

- Operators of regular expression and their precedence
- Algebraic laws for Regular expressions
- Kleen's Theorem
- Regular expression to FA
- Non-Regular Languages

- Pumping Lemma for regular Languages

Module 3: Language & Grammar Formalism [9L]

- Grammars, Regular grammars
- Context-Free Grammar
- Derivation, Derivation trees
- Ambiguity in Grammar
- Simplification of CFGs
- Normal forms for CFGs - CNF and GNF

Module 4: Push Down Automata [4L]

- PDA Description and definition
- Acceptance by Final state, Acceptance by empty stack
- Deterministic PDA
- Equivalence of PDA and CFG

Module 5: Turing Machines and Decidability [6L]

- Basic model, Definition and representation
- Language acceptance by TM
- Church's Thesis
- Recursive and recursively enumerable languages
- Halting problem
- Introduction to Undecidability

Text/Reference Books:

1. J. D. Ullman, J. Hopcroft and R. Motwani, "Introduction to Automata Theory, Languages and Computation", Pearson Education, 2007
2. P. Linz, "An Introduction to Formal Languages and Automata", Jones & Bartlett Learning, 2012
3. K. L. P. Mishra and N. Chandrasekaran, "Theory of Computer Science: Automata, Languages and Computation", Prentice Hall India, 2008
4. M. Sipser, "Introduction to Theory of Computation", Thomson Course Technology, 2006

5. J. C. Martin, “Introduction to Languages and Theory of Computations”, McGraw Hill, 2011

6. E. A. Rich, Automata, “Computability and Complexity”, Pearson Education, Inc., 2019

7. D. Kozen, “Automata and Computability”, Springer, 1997

8. H. R. Lewis and C. H. Papadimitriou, “Elements of the Theory of Computation”, Prentice Hall of India Private Ltd., 1998

9. Z. Kohavi and N. K. Jha, “Switching and Finite Automata Theory”, Cambridge University Press, 2010

10. D. I. A. Cohen, “Introduction to Computer Theory”, John Wiley & Sons, Inc., 1986

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS4003			
Course Title	Object Oriented Programming using Java			
Category	ENGG, Minor			
LTP & Credits	L	T	P	Credits
	3	0	0	3
Total Contact Hours	36			
Pre-requisites	a) Fundamentals of Programming b) Object Oriented Programming			

Learning Objective:

The course aims to provide a comprehensive understanding of Java programming with a focus on object-oriented concepts. Students will learn the fundamentals of Java, including data types, variables, control statements, and arrays. They will also gain knowledge of OOP principles such as encapsulation, inheritance, polymorphism, and abstraction. Additionally, the course covers advanced topics like exception handling, multithreading, and GUI programming in Java.

Course Outcome:

CO1: Explain the process of interaction between objects, classes, and methods.

CO2: Acquire a basic knowledge of Object Orientation with different properties.

CO3: Analyze various string handling functions with various I/O operations.

CO4: Discuss basic code reusability features with respect to Inheritance, package, and Interface.

CO5: Implement Exception handling, Multithreading, and Applet (Web program in Java) programming concepts in Java.

Course Content:

Module 1: Java Basics [8L]

- JVM architecture
- Data types, variables, and operators
- Control statements
- Arrays
- Exception handling
- Java Collections Framework (List, Set, Map, Array List, Linked List, Hash Set, Hash Map)

Module 2: Object-Oriented Programming [7L]

- Encapsulation
- Class fundamentals
- Object & Object reference
- Constructors
- Access control
- Nested methods
- Abstract Class & Interfaces
- Enumerations
- Packages
- Reflection

Module 3: Inheritance and Polymorphism [9L]

- Types of inheritance
- Member access rules

- Usage of this and Super keyword
- Method Overloading and Overriding
- Abstract classes
- Dynamic method dispatch
- Packages and Interfaces
- I/O Streams

Module 4: Exception Handling and Multithreading [6L]

- Exception types
- Try, Catch, Throw, Throws, Finally
- Thread basics
- Thread Life-Cycle
- Synchronizing Threads
- Inter Communication of Threads
- Critical Factor in Thread
- Deadlock

Module 5: GUI Programming and Applet [6L]

- Applet vs. Application
- Applet Lifecycle
- GUI Programming basics
- AWT Components
- Swing Components
- Java Utilities (java.util Package)

Text/Reference Books:

1. "Java: The Complete Reference" by H. Schildt and C. Dann
2. "Programming With Java: A Primer" by E. Balagurusamy
3. "Thinking in JAVA" by B. Eckel
4. "Database Programming with JDBC and JAVA" by G. Reese

CO-POMapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

CourseCode	YBB4001			
CourseTitle	Principles of Management			
Category	HUM, Ability Enhancement Course			
LTP&Credits	L	T	P	Credits
	2	0	0	2
TotalContactHours	24			
Pre-requisites	None			

Learning Objective:

In this course, students will learn about the fundamental principles of management used in the industry and different organizations. They will study various fields of management and related theories, applying them practically to enhance their management skills. By the end of the course, students will understand and interpret the knowledge and skills necessary to work as effective managers.

Course Outcomes:

CO1: To familiarize students with the origins of management principles and compare them with modern trends in management theories.

CO2: To understand the essential functions of management along with theories framed by management experts in the business field.

CO3: To explain the managerial process and related functions that facilitate change.

CO4: To understand the relationship between various levels of management in a business organization and the process of achieving objectives.

CO5: To explain the importance of feedback control in the management process, along with relevant theories, and to apply proper management principles in modern business practices to solve problems.

Course Content:

Module 1: Management [4L]

- Definition, Nature, Importance, and Evolution of Management
- Contributions of Fayol, Taylor, Hawthorne, Maslow

- Management as an Art or Science?
- Functions of a Manager (Duties and Responsibilities)
- Ethics in Management

Module 2: Planning and Control **[4L]**

- Planning (Steps, Types, and Barriers)
- McKinsey Approach, SWOT Analysis
- Operational and Strategic Planning
- Controlling (Concept, Relationship with Planning, Process, Dimensions)
- Management by Objectives (MBO)

Module 3: Decision Making and Organizing **[4L]**

- Decision Making Process
- Certainty and Uncertainty in Decisions
- Brainstorming
- Process of Organizing
- Authority and Responsibility
- Delegation and Empowerment
- Centralization and Decentralization
- Departmentation

Module 4: Staffing **[4L]**

- Manpower Planning
- Job Design
- Selection and Recruitment
- Training and Development
- Performance Appraisal

Module 5: Leadership and Communication **[3L]**

- Role of Leadership
- Theories of Leadership
- Qualities of a Good Leader
- Development of Leadership
- Communication Process and Types
- Electronic Media

Module 6: Group Dynamics **[2L]**

- Group Concept
- Stages of Group Formation
- Types of Groups

Module 7: Recent Trends in Management **[3L]**

- Social Responsibility in Management
- Changes in Management
- Total Quality Management (TQM)
- Stress Management
- International and Global Management
- Crisis Management

Text/Reference Books:

1. H. Cortes, D. S. Bright, and E. Hartman, "Principles of Management".
2. R. B. Rudani, "Principles of Management".
3. M. Gupta, "Principles of Management".
4. L. M. Prasad, "Principles and Practice of Management".

CO-POMapping:

Course Outcome (CO)	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO-1**	**PSO-2**	**PSO-3**
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS4101			
Course Title	Operating Systems Laboratory			
Category	ENGG, Major			
LTP & Credits	L	T	P	Credits
	0	0	3	1.5
Total Contact Hours	36			
Pre-requisites	Data Structures and Algorithms Computer Organization and Architecture			

Learning Objective:

In this laboratory course, students will carry out various software assignments on Unix/Linux shell programming and system calls. Additionally, assignments for simulating important OS modules like CPU scheduling, file system, etc., shall be carried out.

Course Outcome:

CO1: To learn how to write shell scripts.
CO2: To learn how to use Unix/Linux system calls and to design a shell program.
CO3: To analyze the performance of CPU scheduling algorithms through simulation.
CO4: To learn how to use multi-threaded programming.
CO5: To design and implement one OS module like memory management, file system, etc.

Course Content:

Suggestive List of Experiments:

1. Write shell scripts using "bash" shell scripting language for simple system administration tasks, text search and replacement, directory and file manipulation, simple numeric computations, etc.
2. Write programs in C for familiarization with the Unix/Linux system calls fork, exec, wait, exit, dup, pipe, shared memory, etc.
3. Write a command-line interpreter (shell) program using the Unix/Linux system calls with the facilities for: (a) running executable programs, (b) running a program in the background, (c) input and output redirection, (d) command piping.
4. Implementation of various CPU scheduling algorithms in C and compare their performances.
5. Write programs using the "pthread" library with multiple threads and use semaphores for mutual exclusion.
6. Design and implement a Unix-like memory-resident file system using the concept of inodes. OR Implementation of memory management system supporting virtual memory, and analyze the performance.

Text/Reference Books:

1. A. Silberschatz, P. B. Galvin, and G. Gagne, "Operating System Concepts," Wiley Asia.
2. D. M. Dhamdhere, "Operating Systems: A Concept-Based Approach," Tata McGraw-Hill.
3. M. Bach, "Design of the Unix Operating System," Prentice-Hall of India.

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS4102
Course Title	Numerical Methods Laboratory
Category	ENGG, Minor

LTP & Credits	L	T	P	Credits
	0	0	3	1.5
Total Contact Hours	36			
Pre-requisites	C programming			

Learning Objective:

- To provide an overview of numerical techniques applicable to engineering problems.
- To offer insight into the implementation of these techniques.
- To build confidence in tackling numerical solutions.
- To understand the underlying physics behind numerical methods.
- To aid in method selection by understanding how methods work and their limitations.

Course Outcomes:

CO1: Demonstrate proficiency in using computer programming tools for engineering calculations.
CO2: Construct simple computer algorithms using a programming tool.
CO3: Apply simple numerical methods to solve engineering-related mathematical problems.
CO4: Appreciate the limitations and applicability of numerical methods.
CO5: Utilize computer-based numerical methods for solving engineering problems.

Course Content:

Module 1: Solution for Systems of Non-linear Equations

- Roots of non-linear equations using Bisection Method.
- Roots of non-linear equations using Regula Falsi or False Position Method.
- Roots of non-linear equations using Newton-Raphson Method.

Module 2: Solution for Systems of Linear Algebraic Equations

- System of linear equations using Gauss-Elimination Method.
- System of linear equations using Gauss-Seidal Iteration Method.
- System of linear equations using Gauss-Jordan Method.

Module 3: Numerical Integration

- Numerical integration using Trapezoidal Rule.
- Numerical integration using Simpson's Rules.

Module 4: Solution of Interpolation Polynomial

- Solution using Newton Forward Difference Interpolation Polynomial Function.
- Solution using Newton Backward Difference Interpolation Polynomial Function.
- Solution using Lagrange's Interpolation Polynomial Function.

Module 5: Numerical Solution of Ordinary Differential Equations

- Numerical solution of ordinary differential equations by Euler's Method.
- Numerical solution of ordinary differential equations by Runge-Kutta Method of 4th Order.
- Numerical solution of ordinary differential equations by Taylor Series Method.

Text/Reference Books:

1. "Introductory Methods of Numerical Analysis" by S.S. Sastry.
2. "Numerical Methods for Scientific and Engineering Computation" by Mahinder Kumar Jain.
3. "Numerical Methods for Engineers" by Steven C. Chapra and Raymond P. Canale.
4. "Numerical Recipes: Art of Scientific Computing" by W.H. Press, B.P. Flannery, et al.
5. "Numerical Methods" by E. Balagurusamy.
6. "Numerical Methods using MATLAB" by J. M. Mathews and K. Fink.

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS4103			
Course Title	Object Oriented Programming using Java Laboratory			
Category	ENGG, Minor			
LTP & Credits	L	T	P	Credits
	0	0	3	1.5
Total Contact Hours	36			
Pre-requisites	a)Programming Practices I			

Learning Objective:

The learning objective of this course is to provide students with practical experience in implementing object-oriented programming concepts using Java. Through hands-on exercises and projects, students will learn to design and develop Java applications using classes, inheritance, interfaces, packages, exception handling, file handling, multithreading, and event-driven programming. The course aims to enhance students' understanding and proficiency in Java programming, enabling them to solve real-world problems using object-oriented principles.

Course Outcome:

CO1: Implement and demonstrate an understanding of basic object-oriented programming principles such as classes, objects, inheritance, and polymorphism in Java.
CO2: Design and develop Java applications using interfaces, packages, exception handling, and file handling to solve programming problems.
CO3: Develop multithreaded Java applications to achieve concurrent execution of tasks and improve program efficiency.
CO4: Apply event-driven programming concepts in Java to create interactive graphical user interfaces (GUIs) for applications.
CO5: Work on a mini project that integrates various Java programming concepts learned throughout the course to solve a real-world problem.

Course Content:

1. Electricity Bill Generation using Classes [1 day]
2. Currency Converter, Distance Converter, and Time Converter Implementation using Packages

[1 day]

3. Pay Slip Generation using Inheritance [1 day]
4. Stack ADT Implementation using Multiple Inheritance (Interface) [1 day]
5. String Operations using ArrayList [1 day]
6. Abstract Class Implementation [1 day]
7. User-Defined Exception Handling Implementation [1 day]
8. File Handling [1 day]
9. Multithreading Implementation [1 day]
10. Generic Function Implementation [1 day]
11. Calculator Design Using Event-Driven Programming [1 day]
12. Mini Project [1 day]

Text/Reference Books:

1. H. Schildt and C. Dann, "Java: The Complete Reference", McGraw-Hill Education.
2. E. Balagurusamy, "Programming With Java: A Primer", Tata McGraw-Hill.
3. B. Eckel, "Thinking in JAVA", Prentice Hall.
4. G. Reese, "Database Programming with JDBC and JAVA", O'Reilly Media, Inc.

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS4104			
Course Title	Programming Using Python Laboratory			
Category	ENGG, Major			
LTP & Credits	L	T	P	Credits
	0	0	3	1.5
Total Contact Hours	36			
Pre-requisites	Fundamentals of Programming Basic Problem Solving			

Learning Objective:

In this practical course, students will learn Python programming basics and paradigms, including looping, control statements, and string manipulations. They will become familiar with various

modules, packages, and Python libraries used for various applications, such as machine learning and deep learning.

Course Outcome:

CO1: Understand and explain the basic principles of Python programming language and object-oriented concepts.

CO2: Define and demonstrate the use of built-in data structures along with the help of condition checking and looping structures.

CO3: Understand and apply various applications of different modules and packages in Python.

CO4: Learn to handle exceptions and files in Python.

CO5: Apply Python programming concepts to develop a computer game for teaching data structures like stacks and queues, enhancing visualization and understanding of abstract concepts through game-based learning.

Course Content:

1. History, Features, setting up path, working with Python, Basic Syntax, Variable and Data Types, Operator. [1 day]

2. Conditional Statements: If, If-else, Nested if-else, Looping, For, While, Nested loops, Control Statements: Break, Continue, Pass. [1 day]

3. String Manipulation: Accessing Strings, Basic Operations, String slices, Function and Methods. Lists: Introduction, accessing list, Operations, Working with lists, Function and Methods. [2 days]

4. Tuple: Introduction, accessing tuples, Operations, Working, Functions, and Methods. Dictionaries: Introduction, accessing values in dictionaries, Working with dictionaries, Properties. [2 days]

5. Functions: Defining a function, Calling a function, Types of functions, Function Arguments, Anonymous functions, Global and local variables. [1 day]

6. Modules: Importing module, Math module, Random module, Packages, Composition, Input-Output Printing on screen, reading data from the keyboard, Opening and closing file, Reading and writing files, Functions. [2 days]

7. Exception and File Handling: Exception, Exception Handling, except clause, Try & finally clause, User-Defined Exceptions. [1 day]

8. Case study on using a computer game for teaching data structures on stacks and queues. The computer game is developed to help students visualize the data structures and data access operations on stacks and queues. This game-based learning is engaging, fun, and, more importantly, abstract concepts in data structures can be visualized and learned through game-playing. [2 days]

Text/Reference Books:

1. T.R. Padmanabhan, "Programming with Python (1st Ed.)", Springer.

2. R. Thareja, "Python Programming: using Problem Solving Approach (1st Ed.)", Oxford University Press.

3. W. McKinney, "Python Data Analysis (2nd Ed.)", O'Reilly.

CO-PO Mapping:

Course Outcome (CO)	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO-1**	**PSO-2**	**PSO-3**
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Semester 5 Curriculum and Syllabus

Course Code	YCS 5006			
Course Title	Real Time system			
Category	ENGG, Major			
LTP& Credits	L	T	P	Credits
	3	0	0	3
Total Contact Hours	36			
Pre-requisites	Operating System			

Learning Objective:

Real-time operating systems are event-driven and preemptive, meaning the OS can monitor the relevant priority of competing tasks, and make changes to the task priority. Event-driven systems switch between tasks based on their priorities, while time-sharing systems switch the task based on clock interrupts.

Course Outcome:

CO1: Explain the basic concept of Artificial Intelligence and its applications.
CO2: Classify and analyze various AI tools and techniques.
CO3: Learn and evaluate various AI algorithms.
CO4: Apply the basic understanding of artificial intelligence in real-world applications.
CO5: Evaluate and assess the performance, effectiveness, and ethical implications of artificial intelligence systems in various real-world applications.

Course Content:

Module 1: [9L]

Introduction: Introduction to Operating System: Computer Hardware Organization, BIOS and Boot Process, Multi-threading concepts, Processes, Threads, Scheduling

Module 2: [7L]

Basics Of Real-Time Concepts: Terminology: RTOS concepts and definitions, real-time design issues, examples, Hardware Considerations: logic states, CPU, memory, I/O, Architectures, RTOS building blocks, Real-Time Kernel

Module 3: [8L]

Process Management: Concepts, scheduling, IPC, RPC, CPU Scheduling, scheduling criteria, scheduling algorithms Threads: Multi-threading models, threading issues, thread libraries, synchronization Mutex: creating, deleting, prioritizing mutex, mutex internals

Module 4 [8L]

Inter-Process Communication: Messages, Buffers, mailboxes, queues, semaphores, deadlock, priority inversion.

Pipes Memory Management:- Process stack management, run-time buffer size, swapping, overlays, block/page management, replacement algorithms, real-time garbage collection

Module 5: [4L]

Case Studies: Case study Linux POSIX system, RTLinux / RTAI, Windows system, Vxworks, uItron Kernel

Design Issues: structure, process states, data structures, inter-task communication mechanism, Linux Scheduling

Text/Reference Books:

1. J. J Labrosse, "MicroC/OS-II: The Real –Time Kernel", Newnes, 2002.
2. Jane W. S. Liu, "Real-time systems", Prentice Hall, 2000
3. W. Richard Stevens, "Advanced Programming in the UNIX® Environment", 2nd Edition, Pearson Education India, 2011.
4. Philips A. Laplante, "Real-Time System Design and Analysis", 3rd Edition, John Wley& Sons, 2004

CO-PO Mapping:

ourse Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS5002			
Course Title	Database Management System			
Category	ENG,Major			
LTP& Credits	L	T	P	Credits
	3	0	0	3
Total Contact Hours	36			
Pre-requisites	a)Data Structures and Algorithms			

Learning Objective:

In this course, students will learn about data models, conceptualize and depict a database system, design systems using E-R diagrams, learn SQL and relational database design, understand internal storage structures using different file and indexing techniques, and grasp the concepts of transaction processing, concurrency control techniques, and recovery procedures.

Course Outcome:

CO1: Apply E-R diagrams for an application.
CO2: Explain the creation of the normalized relational database model.
CO3: Analyze real-world queries to generate reports.
CO4: Determine whether a transaction satisfies the ACID properties.
CO5: Create and maintain an organization's database.

Course Content:

Module 1: Introduction [3L]

- Concept and overview of DBMS, data models.
- Database languages, database administrator, database users.
- Three-schema architecture of DBMS.

Module 2: Entity-Relationship and Relational Database Model [9L]

- Basic concepts, design issues.
- Mapping constraints, keys, entity-relationship diagram.
- Weak entity sets, extended E-R features.
- Case study on E-R model.
- Structure of relational databases, relational algebra.
- Relational calculus, extended relational algebra operations.
- Views, modifications of the database.

Module 3: SQL and Integrity Constraints [6L]

- Concept of DDL, DML, DCL.
- Basic structure, set operations, aggregate functions.
- Null values, domain constraints, referential integrity constraints.
- Assertions, views, nested sub-queries.
- Database security application development using SQL, stored procedures, and triggers.

Module 4: Relational Database Design [6L]

- Functional dependency, different anomalies in designing a database.
- Normalization using functional dependencies, decomposition, Boyce-Codd Normal Form, 3NF.
- Normalization using multi-valued dependencies, 4NF, 5NF.
- Case study.

Module 5: Internals of RDBMS [6L]

- Physical data structures.
- Query optimization: join algorithm, statistics, and cost-based optimization.
- Transaction processing, concurrency control, and recovery management.
- Transaction model properties, state serializability.
- Lock-based protocols, two-phase locking, deadlock handling.

Module 6: File Organization & Index Structures [6L]

- File and record concept, placing file records on disk.
- Fixed and variable-sized records.
- Types of single-level index (primary, secondary, clustering).
- Multilevel indices, dynamic multilevel indices using B-tree and B+ tree.

Text/ Reference Books:

1. R. Elmasri and S. B. Navathe, "Fundamentals of Database Systems", Addison Wesley Publishing.
2. C. J. Date, "Introduction to Database Management", Vol. I, II, III, AddisonWesley.
3. J. D. Ullman, "Principles of Database Systems", Galgottia Publication.
4. G. Jim and R. Address, "Transaction Processing: Concepts and Techniques", Morgan Kaufmann.

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS 5003			
Course Title	Computer Networks			
Category	ENGG,Major			
LTP& Credits	L	T	P	Credits
	3	0	0	3
Total Contact Hours	36			
Pre-requisites	a) Computer Organization and Architecture b)Operating Systems			

Learning Objective:

In this course, students will learn about the fundamental concepts of computer networking, with a detailed understanding of the TCP/IP protocol suite that drives the Internet. Various important network applications shall be discussed, providing students with a comprehensive understanding of how data flows through a real network and the various issues involved.

Course Outcome:

CO1: Explain the fundamental concepts of data communication.

CO2: Illustrate how the various protocols at the data link layer level work.

CO3: Explain the functionalities of the various protocols at the network and transport layer level.

CO4: Demonstrate how various internetworking devices can be used to connect different networks together.

CO5: Learn about various network applications with a particular emphasis on security.

Course Content:

Module 1: Introduction to Data Communication Techniques [5L]

- Data communication concepts, analog and digital signal transmission.
- Layered Network architecture–the OSI model.
- Transmission media (guided and unguided) and data transmission techniques (analog and digital).
- Signal encoding techniques – NRZ, NRZI, AMI, Manchester, Differential Manchester, etc.
- Circuit switching and packet switching, virtual circuits, and datagrams.

Module 2: Data Link Layer [7L]

- Framing and flow-control techniques.
- Stop-and-wait and sliding-window protocols for frame transmission, performance analysis.
- Error control techniques–checksum and CRC, stop-and-wait ARQ, Go-back-N, selective reject protocols.
- Multiple-access protocols: ALOHA, CSMA and CSMA/CD.
- IEEE 802.x Ethernet standard, switched Ethernet, Fast Ethernet, Gigabit Ethernet.
- Wireless LAN protocols and standards.

Module 3: Network Layer [8L]

- TCP/IP protocol suite, internetworking concepts.
- Internet Protocol (IP), IP addressing and routing, IP fragmentation and reassembly.
- IP subnets and masks–variable length subnet masks, classless inter-domain routing.
- Miscellaneous protocols–ARP and RARP, ICMP, BOOTP and DHCP.
- IPv6–basic differences from IPv4.

Module 4: Transport Layer [4L]

- Process-to-process delivery, TCP and UDP.
- TCP connection establishment and termination.
- Flow and congestion control in TCP–window advertisement, leaky-bucket and token-bucket algorithms.

Module 5: Internetworking Concepts [6L]

- Internetworking devices repeaters, hubs, bridges and routers.
- Interconnecting LANs using bridges, frame forwarding and address learning.
- Routing algorithms–shortest-path algorithm, distance vector algorithm, link state algorithm.
- RIP OSPF and BGP algorithms.

Module 6: Network Applications [6L]

- Client-server concept.
- Introduction to DNS, SMTP, SNMP, FTP, TELNET and HTTP.
- Firewalls, Network Address Translator (NAT), Proxy Server, etc.
- Basic concepts of Cryptography–symmetric and asymmetric key crypto systems, Cryptographic hash functions.

- Digital signature, PGP, HTTPS.

Text/Reference Books:

1. W. Stallings, "Data and Computer Communication (5thEd.)", PHI/Pearson Education.
2. B. A. Forouzan, "Data Communication and Networking (3rdEd.)", Tata-McGraw-Hill.
3. W. R. Stevens, "UNIX Network Programming (3rdEd.), Prentice-Hall, Addision-Wesley.
4. A. Tanenbaum, "Computer Networks (4thEd.), PHI/ Pearson Education.
5. W. Stallings, "Cryptography and Network Security: Principles and Practice(4thEd.)", PHI/Pearson Education.

CO-PO Mapping:

Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS5004A			
Course Title	Compiler Design			
Category	Major			
LTP& Credits	L	T	P	Credits
	3	0	0	3
Total Contact Hours	36			
Pre-requisites	a) Formal Language and Automata Theory b) Computer Organization and Architecture c) Programming and Data Structure			

Learning Objective:

In this course, students will learn about the fundamental principles in compiler design, the algorithms and data structures involved in the construction of a compiler, and automation tools like lex and yacc for translating high-level languages. At the end of the course, students will be able to build different phases of compilers.

Course Outcome:

CO1: Understand the lexical, syntactic, and semantic structures of a language.

CO2: Recall various techniques to modify the grammar of a given language.

CO3: Understand intermediate representations including symbol table, parse/syntax tree, and data structures required for such representations.

CO4: Understand different techniques for intermediate code and machine code optimization.

CO5: Implement various optimization techniques in the compiler design process.

Course Content:

Module 1: Lexical Analysis (8L)

- History of Compiler Design
- Analysis of the Source Program
- The Phases of a Compiler
- Cousins of the Compiler
- The Grouping of Phases
- Compiler Construction Tools
- Need and role of lexical analyzer
- Lexical errors
- Input Buffering
- Specification of Tokens
- Recognition of Tokens
- Design of a Lexical Analyzer Generator
- Use of Lex tool

Module 2: Syntax Analysis (9L)

- Need and role of the parser
- Context-Free Grammars
- Top-Down Parsing
- Recursive Descent Parser
- Predictive Parser
- LL(1) Parser
- Shift Reduce Parser
- LR Parser
- LR Item
- Construction of SLR Parsing table
- Introduction to LALR Parser
- Use of YACC / Bison tool
- Design of a syntax analyzer for a sample language

Module 3: Syntax Directed Translation (7L)

- Syntax-directed Translation schemes
- Implementation of Syntax-directed Translators
- Intermediate code

- Postfix notation
- Parse trees & syntax trees
- Translation of assignment statements
- Boolean expressions
- Statements that alter the flow of control
- Postfix translation
- Translation with a top-down parser
- Translation: Array references in arithmetic expressions, procedure call, declarations, and case statements

Module 4: Code Generation (6L)

- Data structure for symbol tables
- Representing scope information
- Three address code, quadruple & triples
- Issues in the design of code generator
- The target machine
- Runtime Storage management
- Basic Blocks and Flow Graphs
- Next-use Information
- A simple Code generator
- DAG representation of Basic Blocks

Module 5: Code Optimization (6L)

- Sources of Optimization
- Peephole Optimization
- Optimization of basic Blocks
- Introduction to Global Data Flow Analysis
- Runtime Environments
- Source Language issues
- Storage Organization
- Storage Allocation strategies
- Access to non-local names
- Parameter Passing

Text/Reference Books:

1. A. Aho, V.R. Sethi, and D. J. Ullman, "Compilers Principles, Techniques and Tools", Pearson Education.

2. M.L. Scott, "Programming Language Pragmatics", Morgan Kaufmann Publishers.
3. C.N. Fischer, R.K. Cytron, and R.J. LeBlanc, "Crafting a Compiler", Addison-Wesley.
4. S. Chattopadhyay, "Compiler Design", Prentice-Hall of India.
5. A.W. Appel, "Modern Compiler Implementation in C", Cambridge University Press.
6. R. Mark, "Writing Compilers and Interpreters: A Modern Software Engineering Approach Using Java", Wiley Publishing.
7. K.D. Cooper and L. Torczon, "Engineering a Compiler", Morgan Kaufmann Publishers.
8. A.I. Holub, "Compiler Design in C", Prentice-Hall of India.

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS5004B				
Course Title	Cryptography and Network Security				
Category	Major				
LTP & Credits	L	T	P	Credits	
	3	0	0	3	
Total Contact Hours	36				
Pre-requisites	a) Data Structures and Algorithms b) Operating Systems c) Discrete Structures				

Learning Objective:

In this course, students will learn about various cryptographic techniques essential for understanding how secure information systems can be built. The course will discuss various security applications as case studies, helping students strengthen their basic knowledge in cybersecurity.

Course Outcome:

CO1: Explain the basic concept of cryptography and its applications in network security.

CO2: Learn and analyze various private-key cryptography algorithms.

CO3: Learn and analyze various public-key cryptography algorithms.

CO4: Explain various cryptographic hash functions and their applications in network security.

CO5: Demonstrate how the basic concepts of cryptography can be used to develop practical security applications.

Course Content:

Module 1: Introduction to Cryptography and Block Ciphers [4L]

- Introduction to security attacks, services, and mechanisms.
- Conventional encryption models: private-key and public-key cryptography.
- Classical encryption techniques: substitution and transposition ciphers.

Module 2: Private-key Cryptography [7L]

- Block Cipher: Feistel structure, Shannon's theory of confusion and diffusion, DES, triple-DES, AES.
- Linear and differential cryptanalysis: basic concepts.
- Key distribution problem.
- Stream Cipher: basic concept, realization based on linear feedback shift register.

Module 3: Mathematical Background [7L]

- Modular arithmetic, Fermat's and Euler's theorem, gcd, primality testing.
- Euclid's algorithm, Chinese remainder theorem.
- Intractable problems: integer factorization problem, modular square root problem, discrete logarithm problem.

Module 4: Public-key Cryptography [6L]

- RSA algorithm, security of RSA, key management.
- Diffie-Hellman key exchange algorithm.
- Elliptic curve cryptography: basic concepts.

Module 5: Cryptographic Hash Functions and Authentication [5L]

- Properties of hash functions: MD5 message digest algorithm, secure hash algorithm (SHA-1).
- Digital signatures: authentication protocols, various approaches, digital signature standard (DSS).

Module 6: Network Security [7L]

- Authentication applications: Kerberos, X.509 directory authentication service.
- Electronic mail security: pretty good privacy (PGP), S/MIME.
- Certification: public-key infrastructure.
- Secure socket layer (SSL), transport layer security, secure HTTP (HTTPS), and other secure protocols on the Internet.
- System security: viruses, worms, and malware, firewall systems.

Text/Reference Books:

1. W. Stallings, "Cryptography and Network Security: Principles and Practices", Prentice-Hall of India.
2. J. Menezes, P.C. van Oorschot, and S.A. Vanstone, "Handbook of Applied Cryptography", CRC Press.

3. D. Stinson, “Cryptography: Theory and Practice”, CRC Press.
4. C. Kaufman, R. Perlman, and M. Speciner, “Network Security”, Pearson Education.

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS5004C			
Course Title	Computer Graphics			
Category	Major			
LTP& Credits	L	T	P	Credits
	3	0	0	3
Total Contact Hours	36			
Pre-requisites	a. Mathematical Foundation b. Programming Languages			

Learning Objectives:

This course provides an introduction to the principles of computer graphics. In particular, the course will consider methods for modeling 2-D objects and generating photorealistic renderings on color raster graphics devices. The emphasis will be on understanding how various elements like algebra, geometry, algorithms, and data structures interact in the design of graphics. Additionally, this course provides an idea of hardware system architecture for computer graphics, including graphics pipeline, frame buffers, and graphic coprocessors.

Course Outcome:

CO1: Students will describe fundamental algorithms used in computer graphics and compare and evaluate them to some extent.

CO2: Students will work and interact, through hands-on experiences, to design, develop, and modify electronically generated imagery using a wide range of sophisticated graphical tools and techniques.

CO3: Students will summarize different hidden surface elimination algorithms and shading techniques used in computer graphics and digital media production.

CO4: Students will explain the technology necessary for creating multimedia content for the web, video, DVD, 2D and 3D graphics, sound, and programming.

CO5: Students can apply knowledge, techniques, skills, and modern tools to become successful professionals in communication and media industries.

Course Content:

Module 1: Introduction [8L]

- Application areas of Computer Graphics
- Overview of graphics systems
- Video display devices
- Raster-scan systems
- Random scan systems
- Graphics monitors and workstations
- Input devices
- Output primitives: Points and lines
- Line drawing algorithms
- Mid-point circle and ellipse algorithms
- Filled area primitives: Scan line polygon fill algorithm, boundary-fill, and flood-fill algorithms.

Module 2: 2-D Geometrical Transforms [8L]

- Translation, scaling, rotation, reflection, and shear transformations
- Matrix representations and homogeneous coordinates
- Composite transforms
- Transformations between coordinate systems
- 2-D Viewing: The viewing pipeline
- Viewing coordinate reference frame
- Window to view-port coordinate transformation
- Viewing functions
- Cohen-Sutherland clipping algorithms
- Sutherland-Hodgeman polygon clipping algorithm.

Module 3: 3-D Object Representation [6L]

- Polygon surfaces
- Quadric surfaces
- Spline representation
- Hermite curve, Bezier curve, and B-spline curves
- Bezier and B-spline surfaces

- Basic illumination models
- Polygon rendering methods.

Module 4: Geometric Transformations [6L]

- 3-D Geometric transformations: Translation, rotation, scaling, reflection, and shear transformations
- Composite transformations
- 3-D viewing: Viewing pipeline
- Viewing coordinates
- View volume and general projection transforms and clipping.

Module 5: Computer Animation [8L]

- Design of animation sequence
- General computer animation functions
- Raster animation
- Computer animation languages
- Keyframe systems
- Motion specifications
- Visible-Surface Determination Techniques for efficient Visible-Surface Algorithms
- Categories of algorithms
- Back-face removal
- The z-Buffer Algorithm
- Scan-line method
- Painter's algorithms.

Text/Reference Books:

1. "Computer Graphics C version" by Donald Hearn and M. Pauline Baker, Pearson Education.
2. "Computer Graphics Second edition" by Zhigand Xiang, Roy Plastock, Schaum's Outlines, Tata McGraw Hill edition.
3. "Computer Graphics Principles & Practice" (Second edition in C) by Foley, Van Dam, Feiner, and Hughes, Pearson Education.
4. "Procedural Elements for Computer Graphics" (Second edition) by David F. Rogers, Tata McGraw Hill.

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	-	-	2	-	-	-	0	0	3	0	0	0	1	0	0
CO2	2	2	-	-	-	-	0	0	0	0	0	0	1	2	0
CO3	2	-	1	-	3	-	0	0	1	0	0	0	2	0	0
CO4	2	2	-	2	-	1	0	0	1	0	0	0	0	0	0
CO5	1	2	-	-	-	-	0	0	2	0	0	0	1	0	0

Course Code	YED5001			
Course Title	Economics for Engineers			
Category	HUM, Minor			
LTP & Credits	**L**	**T**	**P**	Credits
	2	0	0	2
Total Contact Hours	24			
Pre-requisites	None			

Learning Objective:

In this course, students will learn about managerial economics, basics of accounting, and financial management. By the end of the course, students will be able to make different managerial decisions in terms of economics, solve financial statements, and make different financing decisions for businesses and at a personal level.

Course Outcome:

CO1: Apply appropriate engineering economics analysis methods for problem-solving: present worth, annual cost, rate-of-return, payback, break-even, benefit-cost ratio.

CO2: Evaluate the cost-effectiveness of individual engineering projects using the learned methods and draw inferences for investment decisions.

CO3: Compare the life cycle cost of multiple projects using the learned methods and make a quantitative decision between alternate facilities and/or systems.

CO4: Evaluate the profit of a firm, carry out break-even analysis, and employ this tool to make production decisions.

CO5: Discuss and solve advanced economic engineering analysis problems including taxation and inflation.

Course Content:

Module 1: Introduction [3L]

- Managerial Economics
- Relationship with other disciplines
- Firms: Types, Objectives, and goals
- Managerial Decisions-Decision Analysis

Module 2: Demand and Supply Analysis [5L]

- Demand: Types, determinants, function, elasticity, forecasting
- Supply: Determinants, function, elasticity

Module 3: Cost Analysis [5L]

- Elements of costs
- Marginal cost, Marginal Revenue
- Sunk cost, Opportunity cost
- Break-even analysis–PV ratio

Module 4: Elementary Economic Analysis [4L]

- Inflation: Meaning, types, causes, measures to control
- National Income: Definition, Concepts, Method of measuring

Module 5: Financial Accounting [5L]

- Concepts and Definition of Accounting
- Journal, Ledger, Trial Balance
- Trading A/C, Profit & Loss A /C, Balance Sheet

Module 6: Investment Decision [2L]

- Time value of money: Interest - Simple and compound, nominal and effective rate of interest
- Cash flow diagrams
- Principles of economic equivalence
- Evaluation of engineering projects: Present worth, Future worth, Annual worth, Internal rate of return, Cost-benefit analysis for public projects

Text/Reference Books:

1. B. Riggs and S.U. Randhwa, "Engineering Economics", McGraw Hill Education India.
2. D. Vengedasalam and K. Madhavan, "Principles of Economics", Oxford University Press.
3. W.G. Sullivan, E.M. Wicks, and C.P. Koelling, "Engineering Economy", Pearson.
4. R.P. Seelvan, "Engineering Economics", Prentice-Hall of India.
5. H.L. Ahuja, "Principles of Micro Economics", S. Chand & Company Ltd.
6. S.P. Gupta, "Macro Economics", Tata McGraw Hill.
7. K.K. Dewett, "Modern Economic Theory", S.Chand & Company Ltd.

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS5106			
Course Title	Real Time System Laboratory			
Category	ENGG, Major			
LTP & Credits	L	T	P	Credits
	0	0	3	1.5
Total Contact Hours	36			
Pre-requisites	a) Operating system			

Learning Objective:

An RTOS, or real-time operating system, is a specialized software system designed to manage and prioritize tasks in real-time systems, where tasks have strict timing constraints.

Course Outcomes:

CO1: To provide students a practical experience working on a real-time platform

CO2: To work with Matlab/Simulink.

CO3: To improve C++ and reusability skills.

CO4: To improve programming syntax and algorithm checking.

CO5: To learn QNX and the Integrated Development Environment (IDE).

Course Content:

Suggestive List of Experiments:

1. Create an application that creates two tasks that wait on a timer whilst the main task loops.

2 Write an application that creates a task which is scheduled when a button is pressed, which illustrates the use of an event set between an ISR and a task

3 Write an application that Demonstrates the interruptible ISRs(Requires timer to have higher priority than external interrupt button)

4 a).Write an application to Test message queues and memory blocks.

b).Write an application to Test byte queues

5 Write an application that creates two tasks of the same priority and sets the time slice period to illustrate time slicing.

6 Interfacing Programs: Write an application that creates a two task to Blinking two different LEDs at different timings

7 Write an application that creates a two task to Blinking two different LEDs at different timings

8 Write an application that creates a two task displaying two different messages in LCD display in two lines.

9 Sending messages to mailbox by one task and reading the message from mailbox by another task.

10 Sending message to PC through serial port by three different tasks on priority Basis.

11 Porting Linux and developing simple application on Xilinx Zed board

12 Developing image processing application with Linux OS on Xilinx Zynq FPGA

Text/Reference Books:

1. S.Natarajan, "Real Time Operating System Laboratory Manual", Nats Innovation Factory, ISBN: 978-93-5426-763-5, January2021.

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS5102			
Course Title	Database Management System Laboratory			
Category	ENGG, Major			
LTP & Credits	L	T	P	Credits
	0	0	3	1.5
Total Contact Hours	36			
Pre-requisites	a)Digital Circuits Laboratory			

Learning Objective:

This course aims to enable students to learn about data models, conceptualize and depict a database system, understand the fundamental concepts of SQL queries, design a database with the necessary attributes, and learn the methodology of accessing, modifying, and updating data and information from relational databases. Additionally, students will learn database design, designing user interfaces, and how to connect with databases.

Course Outcome:

CO1:To understand the basic concepts regarding databases and SQL queries.

CO2:To explain the concepts of PL/SQL.

CO3:To differentiate between DBMS and advanced DBMS.

CO4:To analyze database system concepts and apply normalization to the database.

CO5:To apply and create different transaction processing and concurrency control applications.

Course Content:

1. Experiments on fundamentals of database systems:
    - Creating a Database
    - Creating a Table
    - Specifying Relational Data Types
    - Specifying Constraints
    - Creating Indexes (Duration: 2 days)
2. Experiments on database Tables and Record handling:
    - INSERT statement
    - Use of SELECT and INSERT together
    - DELETE, UPDATE, TRUNCATE statements
    - DROP, ALTER statements (Duration: 2 days)
3. Experiments on retrieving data from the database:
    - The SELECT statement
    - Use of the WHERE clause
    - Use of the Logical Operators in the WHERE clause
    - Use of IN, BETWEEN, LIKE, ORDER BY, GROUP BY, and HAVING Clause
    - Use of the Aggregate Functions
    - Combining tables using JOINS
    - Sub-queries (Duration: 3 days)
4. Experiments on Miscellaneous Database Management:
    - Creating Views
    - Creating Column Aliases
    - Creating Database Users
    - Use of GRANT and REVOKE (Duration: 1 day)

5. Experiments on PL/SQL:
    - Use of decision-making statements
    - Different loop structures to solve simple programs (e.g., sum of few numbers, pattern prints, etc.)
    - Inserting values into tables, reading data from a table
    - Basic working with CURSORS (Duration: 1 day)
6. Innovative Experiments:
    - Case study of handling complex databases (e.g., College Management System, Hospital Management System, Library Management System, Payroll Management System, etc.) (Duration: 3 days)

Text/Reference Books:

1. H.F. Korth and A. Silberschatz, "Database System Concepts", McGraw Hill.
2. E. Ramez and S. Navathe, "Fundamentals of Database Systems", Benjamin Cummings Publishing Company.
3. C.J. Date, "Introduction to Database Management", Vol. I, II, III, Addison Wesley.
4. G. Jim and R. Address, "Transaction Processing: Concepts and Techniques", Moragan Kaufmann.
5. J.D. Ullman, "Principles of Database Systems", Galgottia Publication.
6. I. Bayross, "SQL, PL/SQL the Programming Language of Oracle", BPB Publications.

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS5103			
Course Title	Computer Networks Laboratory			
Category	ENGG, Major			
LTP& Credits	L	T	P	Credits
	0	0	3	1.5
Total Contact Hours	36			
Pre-requisites	c. Operating Systems Laboratory d. Computer Organization and Architecture Laboratory			

Learning Objective:

In this laboratory course, students will learn network programming using the socket API system calls and analyze packets flowing over the network. Additionally, they will simulate several algorithms at the data link and network layers and analyze the results.

Course Outcome:

CO1: To learn how to use socket API system calls for network programming.

CO2: To learn how to capture network packets and analyze them.

CO3: To analyze various algorithms at the data link and network layers through simulation.

CO4: To understand and apply error detection and correction techniques in network communication.

CO5: To design and implement simple network protocols.

Course Content:

Suggestive List of Experiments:

1. Familiarization with Berkeley socket interface system calls in C, and writing programs to communicate between two machines using both connection-oriented (TCP) and connectionless (UDP) protocols. [3 days]
2. Write programs in C to simulate the stop-and-wait and sliding-window protocols, and carry out performance analyses both in the absence of errors and also in the presence of errors. [2 days]
3. Familiarization with a packet capturing and analysis tool (like Wireshark), and analyze packets as captured under various data transfer scenarios over the network. [2 days]
4. Write a program in C to simulate a router for filtering packets (make the specification of the problem as realistic as possible). [3 days]
5. Write programs to implement the distance vector algorithm for building up the routing tables in a network of routers. [2 days]

Text/Reference Books:

1. W. Stallings, "Data and Computer Communication (5th Ed.)", PHI/Pearson Education.
2. B. A. Forouzan, "Data Communication and Networking (3rd Ed.)", Tata-McGraw-Hill.
3. W. R. Stevens, "UNIX Network Programming (3rd Ed.)", Prentice-Hall, Addison-Wesley.

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Semester 6 Curriculum and Syllabus

Course Code	YCS6009			
Course Title	Iot Communication Protocols			
Category	ENGG, Major			
LTP & Credits	L	T	P	Credits
	3	0	0	3
Total Contact Hours	36			
Pre-requisites	None			

Learning Objectives:

In this course, learners will be going to learn about various protocols designed for the implementation of the Internet of Things (IoT) applications.

Course Outcomes:

CO1:. Understand fundamentals of IoT architecture outline and standards.
CO2: Understand and analyze different architectural views.
CO3: Understand the importance of IoT Data Link Layer & Network Layer Protocols.
CO4: Understand the importance of Iot Transport & Session Layer Protocols.
CO5: Understand the security aspects and challenges in IoT communication protocols, including authentication, encryption, and secure data transmission.

Course Contents:

Module 1: **[8L]**

Introduction: IoT architecture outline, standards - IoT Technology Fundamentals- Devices and gateways, Local and wide area networking, Data management, Business processes in IoT, Everything as a Service (XaaS), M2M and IoT Analytics.

Module 2: **[7L]**

Iot Reference Architecture: Introduction, Functional View, Information View, Deployment and Operational View, Other Relevant Architectural views. Real-World Design Constraints- Introduction, Technical Design constraints.

Module 3: **[7L]**

IoT Data Link Layer: PHY/MAC Layer (3GPP MTC, IEEE 802.11, IEEE 802.15), Wireless HART, ZWave, Bluetooth Low Energy, Zigbee Smart Energy, DASH7

Module 4: **[6L]**

Network Layer Protocols: Network Layer-IPv4,IPv6, 6LoWPAN, 6TiSCH,ND, DHCP, ICMP, RPL, CORPL, CARP.

Module 5: **[8L]**

IOT Transport & Session Layer Protocols: Transport Layer (TCP, MPTCP, UDP, DCCP, SCTP)-(TLS, DTLS) – Session Layer-HTTP, CoAP, XMPP, AMQP, MQTT.

Text/ Reference Books:

1. Daniel Minoli, “Building the Internet of Things with IPv6 and MIPv6: The Evolving World of M2M Communications”, ISBN: 978-1-118-47347-4, Willy Publications ,2016

2. Jan Holler, Vlasios Tsiatsis, Catherine Mulligan, Stefan Avesand, Stamatis Karnouskos, David Boyle, "From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence",1st Edition, Academic Press, 2015.
3. Bernd Scholz-Reiter, Florian Michahelles, "Architecting the Internet of Things", ISBN 978-3-642-19156-5 e-ISBN 978-3-642-19157-2, Springer, 2016.
4. N. Ida, Sensors, Actuators and Their Interfaces, Scitech Publishers, 2014.

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS6010			
Course Title	Embedded System			
Category	Major ENGG			
LTP & Credits	L	T	P	Credits
	3	0	0	3
Total Contact Hours	36			
Pre-requisites	None			

Learning Objectives:

For the development of the small-scale embedded system, we need an integrated development environment (IDE) for writing the code). An embedded system is a combination of computer hardware and software designed for a specific function. The purpose of embedded systems is to control a specific function within a device.

Course Outcomes:

CO1: Explain the fundamental concepts and history of embedded systems, including their classifications, applications, and purposes.
CO2: Identify and compare different types of microprocessors and microcontrollers used in embedded systems, including their architectures and characteristics.
CO3: Demonstrate the ability to program embedded systems using C and C++, including compiling, linking, locating, downloading, and debugging.
CO4: Describe the hardware components of embedded systems, including memory maps, I/O maps, interrupt maps, external peripherals, and memory types.
CO5: Explain the role of peripherals, device drivers, and timers in embedded systems, and evaluate the selection process for embedded operating systems based on real-time characteristics.

Course Contents:

Module 1: **[7L]**

Introduction: Embedded Systems and general purpose computer systems, history, classifications, applications and purpose of embedded systems. Core of Embedded Systems : Microprocessors and microcontrollers, RISC and CISC controllers, Big endian and Little endian processors, Application specific ICs, Programmable logic devices,

COTS, sensors and actuators, communication interface, embedded firmware, other system components, PCB and passive components.

Module 2: **[6L]**

Characteristics and quality attributes of embedded systems : Characteristics, Operational and nonoperational quality attributes, application specific embedded system - washing machine, domain specific -automotive.

Module 3: **[7L]**

Programming Embedded Systems : Structure of embedded program, infinite loop, compiling, linking and locating, downloading and debugging.

Module 4: **[6L]**

Embedded hardware : Memory map, i/o map, interrupt map, processor family, external peripherals, memory -RAM, ROM, types of RAM and ROM, memory testing, CRC, Flash memory.

Module 5: **[5L]**

Peripherals : Control and Status Registers, Device Driver, Timer Driver-Watchdog Timers, Embedded Operating System, Real-Time Characteristics, Selection Process.

Module 6: **[5L]**

Design and Development: Embedded System development environment - IDE, Types of file generated on cross compilation, disassemble / decompile, simulator, emulator and debugging, embedded product development life-cycle, trends in embedded industry.

Text / Reference Books:

1. Programming Embedded Systems in C and C++, First Edition January, Michael Barr, O' Reilly
2. Introduction to embedded systems, Shibu K V Tata McGraw-Hill.
3. Embedded Systems, Rajkamal, TataMcGraw-Hill

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS6011			
Course Title	Programming language for IoT			
Category	Major ENGG			
LTP & Credits	L	T	P	Credits
	3	0	0	3
Total Contact Hours	36			
Pre-requisites	None			

Learning Objective:

This program aims to train students to be equipped with a solid theoretical foundation, systematic professional knowledge and strong practical skills in the Raspberry Pi.

Course Outcome:

CO1: Appreciate the development technology for IoT.
CO2: Familiar with Basic Concepts of Linux.
CO3: Design real time IoT Devices and Familiar with basic foundations of Python Programming and libraries
CO4: Comprehend the basic concepts of Mobile Cloud Computing.
CO5: Analyze and implement IoT solutions using Raspberry Pi, Python programming, and relevant libraries, demonstrating proficiency in integrating hardware and software components for IoT applications.

Course content:

Module 1: **[8L]**

Getting Started with Raspberry Pi: Basic functionality of Raspberry Pi B+ board, setting up the board, configuration and use, booting Raspberry Pi 3, Downloading an Operating System, format an SD card and booting the OS, Interfacing Hardware with the Raspberry Pi, Raspberry Pi Remote Access, operates the Raspberry Pi in "headless mode", Bash Command line, operating Raspberry Pi without needing a GUI interface. Basics of Python programming language: Programming on the Raspberry Pi. Python on Raspberry Pi, Python Programming Environment, Python Expressions, Strings, Functions and Function arguments, Lists, List Methods, Control Flow.

Module 2: **[8L]**

Introducing Micro Python: Micro Python Features, Micro Python Limitations, Experimenting with Python on PC, Installing Python 3 on Windows 10, Running the Python Console, Running Python Programs with the Interpreter, The Run, Evaluate, Print Loop (REPL Console), Off and Running with MicroPython, Additional Hardware, Basic Electronics Kit, Breadboard and Jumper Wires and 3 Examples.

Module 3: **[6L]**

IoT Physical Servers and Cloud Offerings: Introduction to Cloud Storage models and communication APIs. Web Server – Web server for IoT, Cloud for IoT, Python web application framework. Designing a REST ful web API. Connecting to APIs.

Module 4: **[7L]**

Baking Pi: Powering Raspberry Pi, Formatting SD cards, Installing and connecting Raspberry pi, How to tell Raspberry pi is working, Installing Raspbian with NOOBS, Networking Raspberry Pi, Connecting with Ethernet, Connecting Via Local Computer Network, Connecting Via Wireless Network, Updating and Upgrading, Setting up a Host Name, Connecting Raspberry pi with SSH, Creating Simple Rasspberry pi application.

Module 5: **[7L]**

FIRST Project on Java: Bill of Materials, Getting Started with Net Beans, Downloading and Configuring Net Beans, Revisiting Hello Raspberry Pi, Brewing Java, Communicating with a USB Scale, Coffee Calculator, Asynchronous Communication, Coffee Brewing Recipe, Commercial Licensing.

Text/ Reference Books:

1. Simon Monk, "Programming the Raspberry Pi: Getting Started with Python", January 2012, McGraw Hill Professional.

2. MicroPython for the Internet of Things, A Beginner's Guide to Programming with Python on Microcontrollers, Charles Bell, Apress.
3. Raspberry Pi with Java: Programming the Internet of Things (IoT) (Oracle Press) 1st Edition.
4. Eben Upton and Gareth Halfacree, "Raspberry Pi User Guide", August 2016, 4th edition, John Wiley & Sons
5. Alex Bradbury and Ben Everard, "Learning Python with Raspberry Pi", Feb 2014, JohnWiley & Sons
6. Michael Margolis, "Arduino Cookbook", First Edition, March 2011, O'Reilly Media, Inc

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS6004C			
Course Title	Cloud Computing			
Category	Open Elective			
LTP & Credits	L	T	P	Credits
	3	0	0	3
Total Contact Hours	36			
Pre-requisites	a) Computer Networks b) Operating Systems			

Learning Objective:

To provide students a sound foundation of cloud computing so that they are able to start using and adopting Cloud Computing services and tools in their real-life scenarios. To enable students to explore some important cloud computing-driven commercial systems and applications.

Course Outcome:

CO1: To explain the Cloud architecture, different services, and deployment models.
CO2: To learn the concepts of abstraction and different types of virtualization.
CO3: To identify and explain different cloud offerings with their usage, namely Azure, Google Apps, Amazon Web Services.
CO4: To explain the underlying concepts of cloud management and security and illustrate the use of Service -Oriented Architecture (SOA).
CO5: To analyze and compare different cloud computing providers and services, evaluating their performance, cost-effectiveness, and suitability for specific use cases.

Course Content:

Module 1: Definition of Cloud Computing and its Basics [10L]

Definition of Cloud Computing: Defining a Cloud, Cloud Types – NIST model, Cloud Cube model, Deployment models (Public, Private, Hybrid, and Community Clouds), Service models –

Infrastructure as a Service, Platform as a Service, Software as a Service with examples of services/ service providers, Cloud Reference model Characteristics of Cloud Computing – a shift in paradigm Benefits and advantages of Cloud Computing.

- Cloud Architecture: A brief introduction to Composability, Infrastructure, Platforms, Virtual
- Appliances, Communication Protocols, Applications, Connecting to the Cloud by Clients.
- Services and Applications by Type IaaS – Basic concept, Workload, partitioning of virtual private server instances, Pods, aggregations, silos.
- PaaS – Basic concept, tools, and development environment with examples.
- SaaS - Basic concept and characteristics, Open SaaS and SOA, examples of SaaS platforms.
- Identity as a Service (IDaaS).
  Compliance as a Service (CaaS).

Module 2: Use of Platforms in Cloud Computing [9L]

- Concepts of Abstraction and Virtualization.
- Virtualization technologies: Types of virtualization (access, application, CPU, storage), Mobility patterns (P2V, V2V, V2P, P2P, D2C, C2C, C2D, D2D).
- Load Balancing and Virtualization: Basic Concepts, Network resources for load balancing, Advanced load balancing (including Application Delivery Controller and Application Delivery Network), Mention of The Google Cloud as an example of the use of load balancing.
- Hypervisors: Virtual machine technology and types, VMware vSphere Machine Imaging (including mention of Open Virtualization Format – OVF).
- Porting of applications in the Cloud: The simple Cloud API and AppZero Virtual Application appliance.

Module 3: Use of various Web Services [5L]

- Concepts of Platform as a Service: Definition of services, Distinction between SaaS and PaaS (knowledge of Salesforce.com and Force.com).
- Application development Use of PaaS Application frameworks.
- Use of Google Web Services: Discussion of Google Applications Portfolio – Indexed search, Dark
- Web, Aggregation and disintermediation, Productivity applications and service, Adwords, Google Analytics, Google Translate, a brief discussion on Google Toolkit (including introduction of Google APIs in brief), major features of Google App Engine service.
- Use of Amazon Web Services: Amazon Web Service components and services: Amazon Elastic Cloud, Amazon Simple Storage system, Amazon Elastic Block Store, Amazon SimpleDB and Relational Database Service.
- Use of Microsoft Cloud Services: Windows Azure platform: Microsoft's approach, architecture, and main elements, overview of Windows Azure AppFabric, Content Delivery Network, SQL Azure, and Windows Live services.

Module 4: Cloud Infrastructure [6L]

- Types of services required in implementation – Consulting, Configuration, Customization, and Support.
- Cloud Management: An overview of the features of network management systems and a brief introduction of related products from large cloud vendors.
- Monitoring of an entire cloud computing deployment stack – an overview with mention of some products.
- Lifecycle management of cloud services (six stages of lifecycle).
- Concepts of Cloud Security: Cloud security concerns, Security boundary, Security service boundary Overview of security mapping Security of data: Brokered cloud storage access, Storage location and tenancy, encryption, and auditing and compliance Identity management (awareness of Identity protocol standards).

Module 5: Concepts of Services and Applications [6L]

- Service Oriented Architecture: Basic concepts of message-based transactions, Protocol stack for an SOA architecture, Event-driven SOA, Enterprise Service Bus, Service catalogs.
- Applications in the Cloud: Concepts of cloud transactions, functionality mapping, Application attributes, Cloud service attributes, System abstraction and Cloud Bursting, Applications and Cloud APIs.
- Cloud-based Storage: Cloud storage definition – Manned and Unmanned Webmail Services: Cloud mail services including Google Gmail, Mail2Web, Windows Live Hotmail, Yahoo mail, concepts of Syndication services.

Text/Reference Books:

1. B. Sosinsky, "Cloud Computing Bible (1st Ed.)", Wiley.

2. R. Buyya, C. Vecchiola, S. T. Selvi, "Mastering Cloud Computing (2nd Ed.)", McGraw Hill Education.
3. A.T. Velte, "Cloud computing: A practical approach (3rd Ed.)", Tata McGraw Hill.
4. C. Miller, "Cloud Computing (4th Ed.)", PHI/Pearson Education.
5. K. Saurabh, "Cloud Computing (2nd Ed.)", Wiley.

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS6004F			
Course Title	Software Engineering			
Category	Major ENGG			
LTP & Credits	L	T	P	Credits
	3	0	0	3
Total Contact Hours	36			
Pre-requisites				

Learning Objective:

The students will be able to comprehend the various software process models. To understand the types of software requirements and SRS document. To know the different software design and architectural styles. To learn the software testing approaches and metrics used in software development. To know about quality control and risk management.

Course Outcome:

CO1: To compare and select a process model for a business system.
CO2: To identify and specify the requirements for the development of an application.
CO3: To develop and maintain efficient, reliable and cost effective software solutions.
CO4: To critically think and evaluate assumptions and arguments of the client.
CO5: To understand the importance of software maintenance and the techniques involved in maintaining software systems.

Course Content:

Module 1: **[9L]**

Introduction to Software Engineering: The evolving role of software, Changing Nature of Software, Software myths. A Generic view of process: Software engineering- A layered technology, a process framework, Process patterns, process assessment. Process models: The waterfall model, Incremental process models, Evolutionary process models, The Unified process, Agility and Agile Process model, Extreme Programming, Other process models of Agile Development and Tools

Module 2: **[8L]**

Software Requirements: Functional and non-functional requirements, User requirements, System requirements, Interface specification, the software requirements document. Requirements Engineering process: Feasibility studies, Requirements elicitation and analysis, Requirements validation, Requirements management. System models: Context Models, Behavioral models, Data models, Object models, structured methods. UML Diagrams.

Module 3: **[7L]**

Design Engineering: Design process and Design quality, Design concepts, the design model. Creating an architectural design: Software architecture, Data design, Architectural styles and patterns, Architectural Design. Object-Oriented Design: Objects and object classes, An Object-Oriented design process, Design evolution. Performing User interface design: Golden rules, User interface analysis and design, interface analysis, interface design steps, Design evaluation

Module 4: **[6L]**

Testing Strategies: A strategic approach to software testing, test strategies for conventional software, Black -Box and White-Box testing, Validation testing, System testing, the art of Debugging. Product metrics: Software Quality, Metrics for Analysis Model, Metrics for Design Model, Metrics for source code, Metrics for testing, Metrics for maintenance. Metrics for Process and Projects: Software Measurement, Metrics for software quality.

Module 5: **[6L]**

Risk management: Reactive vs. Proactive Risk strategies, software risks, Risk identification, Risk projection, Risk refinement, RMMM, RMMM Plan. Quality Management: Quality concepts, Software quality assurance, Software Reviews, Formal technical reviews, Statistical Software quality Assurance, The Capability Maturity Model Integration (CMMI), Software reliability, The ISO 9000 quality standards.

Text/Reference Books:

1. Software Engineering A practitioner's Approach, Roger S Pressman, 6th edition. McGraw Hill International Edition.
2. Software Engineering, Ian Sommerville, 7th edition, Pearson education.
3. Software Engineering, A Precise Approach, Pankaj Jalote, Wiley India, 2010.
4. Software Engineering: A Primer, Waman S Jawadekar, Tata McGraw-Hill, 2008
5. Software Engineering, Principles and Practices, Deepak Jain, Oxford University Press.
6. Software Engineering1: Abstraction and modelling, Diner Bjorner, Springer International edition, 2006.

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS6004G			
Course Title	I.T Security Compliance and Forensic			
Category	Major ENGG			
LTP & Credits	L	T	P	Credits
	3	0	0	3
Total Contact Hours	36			
Pre-requisites	None			

Learning Objectives:

Security consultant , Investigations consultant…Here are the essential objectives of using Computer forensics: It helps to recover, analyze, and preserve computer and related materials in such a manner that it helps the investigation agency to present them as evidence in a court of law.

Course Outcomes:

CO1: Understand the fundamentals of information security and best practices.
CO2: Analyze ethical considerations and legal aspects in cybersecurity.
CO3: Perform penetration testing and identify vulnerabilities in web applications.
CO4: Conduct forensic investigations and ensure network assurance.
CO5: Implement information risk management strategies and respond to cyber incidents.

Course Contents:

Module 1:

Introduction to Information Security Fundamentals and Best Practices: Protecting Your Computer and its Contents, Securing Computer Networks—Basics of Networking, Compromised Computers, Secure Communications and Information Security Best Practices, Privacy Guidelines, Safe Internet Usage.

Module 2:

Ethics in Cyber Security & Cyber Law: Privacy, Intellectual Property, Professional Ethics, Freedom of

Speech, Fair User and Ethical Hacking, Trademarks, Internet Fraud, Electronic Evidence, Cybercrimes.

Module 3:

Penetration Testing: Overview of the web from a penetration testers perspective, Exploring the various servers and clients, Discussion of the various web architectures, Discussion of the different types of vulnerabilities, Defining a web application test scope and process, Defining types of penetration testing.

Module 4:

Web Application Security: Common Issues in Web Apps, What is XSS, SQL injection, CSRF, Password Vulnerabilities, SSL, CAPTCHA, Session Hijacking, Local and Remote File Inclusion, Audit Trails, Web Server Issues.

Module 5:

Forensics & Network Assurance: Forensic Technologies, Digital Evidence Collection, Evidentiary Reporting, Layered Defense, Surveillance and Reconnaissance, Outsider Thread Protection

Module 6:

Information Risk Management: Asset Evaluation and Business Impact Analysis, Risk Identification, Risk Quantification, Risk Response Development and Control, Security Policy, Compliance, and Business Continuity. Forensic investigation using Access Data FTK, En-Case

Module 7:

Cyber Incident Analysis and Response: Incident Preparation, Incident Detection and Analysis. Containment, Eradication, and Recovery. Proactive and Post-Incident Cyber Services, CIA triangle

Text / Reference Books:

1. The Official CHFI Study Guide for Computer Hacking Forensic Investigator by Dave Kleiman
2. CISSP Study Guide, 6th Edition by James M. Stewart

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS6012			
Course Title	Industrial IoT			
Category	Major ENGG			
LTP & Credits	**L**	**T**	**P**	Credits
	3	0	0	3
Total Contact Hours	36			
Pre-requisites	None			

Learning Objectives:

To provide students with good depth of knowledge of Designing Industrial IOT Systems for various application. Knowledge for the design and analysis of Industry 4.0Systems for Electronics Engineering students

Course Outcomes:

At the end of this course, students will be able to:
CO1: Knowledge of theory and practice related to Industrial IoT Systems.
CO2: Ability to identify, formulate and solve engineering problems by using Industrial IoT.
CO3: Ability to implement real field problem by gained knowledge of Industrial applications with IoT capability.
CO4: Ability to design and implement secure communication protocols for Industrial IoT systems, considering the unique challenges of industrial environments such as reliability, latency, and security.
CO5: Understanding of ethical considerations and regulatory requirements related to Industrial IoT systems, and ability to integrate these considerations into the design process.

Course Contents:

Module 1: **[6L]**

Introduction to Industrial IoT (IIoT) Systems: The Various Industrial Revolutions, Role of Internet of Things (IoT) & Industrial Internet of Things (IIoT) in Industry, Industry 4.0 revolutions, Support System for Industry 4.0, Smart Factories.

Module 2: **[6L]**

Implementation systems for IoT: Sensors and Actuators for Industrial Processes, Sensor networks, Process automation and Data Acquisitions on IoT Platform, Microcontrollers and Embedded PC roles in IIoT, Wireless Sensor nodes with Bluetooth, WiFi, and LoRa Protocols and IoT Hub systems

Module 3: **[6L]**

IIoT Data Monitoring & Control: IoT Gate way, IoT Edge Systems and It's Programming, Cloud computing, Real Time Dashboard for Data Monitoring, Data Analytics and Predictive Maintenance with IIoT technology.

Module 4: **[6L]**

Cyber Physical Systems: Next Generation Sensors, Collaborative Platform and Product Lifecycle Management, Augmented Reality and Virtual Reality, Artifical Intelligence, Big Data and Advanced Analysis

Module 5: **[6L]**

Industrial IoT- Applications: Healthcare, Power Plants, Inventory Management & Quality Control, Plant Safety and Security (Including AR and VR safety applications), Facility Management

Module 6: **[6L]**

Case Studies of IIoT Systems: IIoT application development with Embedded PC based development boards, Development of mini Project on new version of Operating systems and Edge development board. That project should also address to the current societal needs

Text / Reference Books:

1. Industry 4.0: The Industrial Internet of Things Alasdair GilchristPublications: Apress
2. The Concept Industry 4.0 An Empirical Analysis of Technologies and Applications in Production Logistics Authors: Bartodziej, Christoph Jan Springer: Publication in the field of economic science.
3. 3. Embedded System: Architecture, Programming and Design by Rajkamal, TMH3.
4. Dr. OvidiuVermesan, Dr. Peter Friess, "Internet of Things: Converging Technologies for Smart Environments and Integrated Ecosystems", River Publishers

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS6109			
Course Title	IoT Communication Protocol Laboratory			
Category	Major ENGG			
LTP & Credits	L	T	P	Credits
	0	0	3	1.5
Total Contact Hours	36			
Pre-requisites	None			

Learning Objectives:
Graduates will be enrolled in academic program in the disciplines of electrical engineering or other disciplines. Graduates will be employed as productive and valued engineers in reputed industries. Graduates will assume increasingly responsible positions and use the technical skills and analytical acumen to address professional values, ethics, leadership, and team skills for execution of complex technological solutions.

Course Outcomes:

At the end of this course, students will be able to:
CO1: Demonstrate knowledge in designing complex energy efficient embedded systems
CO2: Analyze usage of various on-chip resources like GPIO, Timers, Interrupts, ADC, DAC, Comparator, SPI.
CO3: Design embedded systems to suit market requirements.
CO4: Solve engineering problems by proposing potential solutions using industry choice advanced Microcontrollers
CO5: Provide embedded system solutions for societal needs

Course Contents:

List of Experiments:

1. **Introduction to MQTT Protocol**

Setting up an MQTT broker (e.g., Mosquitto)
Publishing and subscribing to MQTT topics using MQTT clients (e.g., MQTT.fx)

2. **MQTT Broker Configuration**

Configuring security settings (e.g., username/password, TLS)
Exploring QoS levels and their impact on message delivery

3. **IoT Device Integration**

Connecting IoT devices (e.g., Raspberry Pi, ESP32) to an MQTT broker
Publishing sensor data (e.g., temperature, humidity) to MQTT topics

4. **MQTT Communication**

Implementing basic publish-subscribe interactions
Testing message delivery under different network conditions

5. **MQTT Security**

Implementing MQTT over TLS/SSL
Testing secure MQTT connections with client authentication

6. **CoAP Protocol Basics**

Setting up a CoAP server (e.g., Californium)
Sending CoAP requests and handling CoAP responses

7. **Resource Observation with CoAP**

Implementing CoAP resource observation
Monitoring changes in CoAP resources

8. **Comparative Analysis: MQTT vs. CoAP**

Comparing MQTT and CoAP in terms of overhead, message format, and suitability for different IoT applications

9. **HTTP RESTful Communication**

Interfacing IoT devices with HTTP-based REST APIs
Sending sensor data via HTTP POST requests

10. **WebSocket Protocol**

Setting up a WebSocket server (e.g., Node.js)
Establishing WebSocket connections from IoT devices

11. **WebSocket for Real-Time Data**

Sending real-time sensor data over WebSocket connections
Visualizing data on a web-based dashboard using WebSocket clients

12. **IoT Device Management**

Implementing OTA (Over-the-Air) updates for IoT devices
Using MQTT or CoAP for remote device management (e.g., firmware updates)

13. **IoT Protocol Performance Analysis**

Measuring latency and throughput of MQTT, CoAP, and HTTP protocols
Analyzing protocol efficiency under different network loads

14. **Fault Tolerance and Reliability**

Simulating network disruptions (e.g., intermittent connectivity)
Evaluating how MQTT and CoAP handle message delivery in unreliable networks

15. **IoT Protocol Integration**

Integrating multiple IoT protocols (e.g., MQTT, CoAP) in a single application scenario
Demonstrating interoperability between different IoT devices and protocols

Text / Reference Books:

1. Daniel Minoli, "Building the Internet of Things with IPv6 and MIPv6: The Evolving World of M2M Communications", ISBN: 978-1-118-47347-4, Willy Publications ,2016
2. Jan Holler, Vlasios Tsiatsis, Catherine Mulligan, Stefan Avesand, Stamatis Karnouskos, David Boyle, "From Machine-to-Machine to the Internet of Things: Introduction to a New Age of Intelligence",1st Edition, Academic Press, 2015.
3. Bernd Scholz-Reiter, Florian Michahelles, "Architecting the Internet of Things", ISBN 978-3-642-19156-5 e-ISBN 978-3-642-19157-2, Springer, 2016.
4. N. Ida, Sensors, Actuators and Their Interfaces, Scitech Publishers, 2014.

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS6110			
Course Title	Embedded System Laboratory			
Category	Major ENGG			
LTP & Credits	L	T	P	Credits
	0	0	3	1.5
Total Contact Hours	36			
Pre-requisites	None			

Learning Objectives:
The main objective of embedded system is to impart the knowledge of Microcontroller based system design and real time system to meet the today's demands of the Industry as well as Research Groups.

Course Outcomes:
At the end of this course, students will be able to:
CO1: Demonstrate knowledge in designing complex energy efficient embedded systems
CO2: Analyze usage of various on-chip resources like GPIO, Timers, Interrupts, ADC, DAC, Comparator, SPI.
CO3: Design embedded systems to suit market requirements.
CO4: Solve engineering problems by proposing potential solutions using industry choice advanced Microcontrollers
CO5: Provide embedded system solutions for societal needs

Course Contents:

List of Experiments:

1. Introduction to MSP430 launch pad and Programming Environment.
2. Read input from switch and Automatic control/flash LED (soft-ware delay).
3. Interrupts programming example using GPIO.
4. Configure watchdog timer in watchdog & interval mode.
5. Configure timer block for signal generation (with given frequency).
6. Read Temperature of MSP430 with the help of ADC.
7. Test various Power Down modes in MSP430.
8. PWM Generator.
9. Use Comparator to compare the signal threshold level.

10. Speed Control of DC Motor
11. Master slave communication between MSPs using SPI.
12. Networking MSPs using Wi-Fi.

TOOL REQUIREMENT: Code Composer Studio Version 6, MSP430 based launch pads, Wi-Fi booster pack.

Text / Reference Books:

1. John H Davies, MSP430 Microcontrollers Basics, Newnes Pub-lishers, 1st edition, 2008.
2. C P Ravikumar, MSP430 Microcontrollers in Embedded System Projects, Elite Publishing House , 1stedition, 2012.

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS6111			
Course Title	Programming language for IoT Laboratory			
Category	Major ENGG			
LTP & Credits	L	T	P	Credits
	0	0	3	1.5
Total Contact Hours	36			
Pre-requisites	None			

Learning Objectives:
Coding serves as the backbone that brings IoT devices to life, enabling them to communicate, collect data, and perform intelligent tasks.

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

CO1: Describe the significance of coding in IoT applications.
CO2: Implement Python scripts to control GPIO pins and read analog sensor data.
CO3: Establish serial communication between microcontrollers and Raspberry Pi using Python.
CO4: Develop MQTT publisher and subscriber scripts for IoT applications.
CO5: Create IoT data visualization tools and simple control systems using Python.

Course Contents:
List of Experiments:

1. Python and GPIO Interfacing
   Writing Python scripts to control GPIO pins on a Raspberry Pi or similar board.
   Blinking an LED or reading a button press using GPIO.
2. Analog Sensor Interfacing
   Using Python to read analog sensor data (e.g., temperature, light intensity) via
   ADC (Analog-to-Digital Converter) modules.
3. Serial Communication

Implementing serial communication between a microcontroller (e.g., Arduino) and Raspberry Pi using Python. Sending and receiving data via UART or USB serial.

4. MQTT Publisher
   Developing a Python script to publish sensor data to an MQTT broker (e.g., Mosquitto). Sending temperature or humidity readings to specific MQTT topics.
5. MQTT Subscriber
   Writing a Python script to subscribe to MQTT topics and receive sensor data. Processing incoming MQTT messages and displaying or logging the data.
6. IoT Data Visualization
   Integrating Python with visualization libraries (e.g., Matplotlib, Plotly) to plot real-time sensor data fetched from MQTT.
7. IoT Control System
   Building a simple IoT control system using MQTT.
   Using Python scripts to control actuators (e.g., motors, relays) based on MQTT messages.
8. IoT Security
   Implementing secure MQTT communication using TLS/SSL in Python.
   Configuring client-side certificates for authentication.
9. Web-based IoT Dashboard
   Creating a web-based IoT dashboard using Python web frameworks (e.g., Flask, Django). Displaying sensor data fetched from MQTT or RESTful APIs.
10. RESTful API Development
    Building a RESTful API server in Python (using Flask or FastAPI) to receive and respond to IoT device requests.
    Sending sensor data to the API and retrieving it from IoT devices.
11. IoT Device Simulation
    Simulating IoT devices using Python scripts.
    Emulating sensor readings and MQTT communication for testing purposes.
12. IoT Edge Computing
    Implementing edge computing tasks (e.g., data preprocessing, filtering) using Python on IoT devices like Raspberry Pi.

Text / Reference Books:

1. Simon Monk, “Programming the Raspberry Pi: Getting Started with Python”, January 2012, McGraw Hill Professional.
2. MicroPython for the Internet of Things, A Beginner’s Guide to Programming with Python on Microcontrollers, Charles Bell, Apress.
3. Raspberry Pi with Java: Programming the Internet of Things (IoT) (Oracle Press) 1st Edition.
4. Eben Upton and Gareth Halfacree, “Raspberry Pi User Guide”, August 2016, 4th edition, John Wiley & Sons
5. Alex Bradbury and Ben Everard, “Learning Python with Raspberry Pi”, Feb 2014, JohnWiley & Sons
6. Michael Margolis, “Arduino Cookbook”, First Edition, March 2011, O'Reilly Media, Inc

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Semester 7 Curriculum and Syllabus

Course Code	YCS7006			
Course Title	IOT cloud processing and Analytics			
Category	ENGG Major			
LTP & Credits	L	T	P	Credits
	3	1	0	4
Total Contact Hours	48			
Pre-requisites	None/If Any			

Learning Objective:
This course covers fundamental concepts, practical applications, and advanced topics related to IoT cloud processing and analytics, providing students with a comprehensive understanding of the subject.

Course Outcome:
CO1: Understand the fundamental concepts of IoT architecture, including sensors, communication protocols, and data transmission mechanisms.
CO2: Develop skills in data analytics for IoT, including data preprocessing, statistical analysis, and visualization of IoT data.
CO3: Learn about big data technologies and frameworks for handling large volumes of IoT data, both in batch and streaming modes.
CO4: Gain insights into security and privacy challenges in IoT cloud processing and acquire skills in implementing security mechanisms and best practices.
CO5: Identify and discuss emerging trends and future directions in IoT cloud processing, including ethical considerations and societal impacts.

Course Content:

Module 1: Introduction to IoT : **[6L]**

Overview of IoT architecture and components, IoT communication protocols (MQTT, CoAP, HTTP), IoT devices and sensors, IoT data collection and transmission

Module 2: Cloud Computing Basics : **[6L]**

Introduction to cloud computing, Cloud service models (IaaS, PaaS, SaaS), Cloud deployment models (Public, Private, Hybrid), Cloud infrastructure and virtualization

Module 3: IoT Cloud Platforms: **[6L]**

Overview of IoT cloud platforms (AWS IoT, Azure IoT, Google Cloud IoT), Setting up IoT devices on cloud platforms, Managing IoT devices and data streams, Security considerations in IoT cloud platforms.

Module 4: Data Analytics for IoT : **[4L]**

Introduction to data analytics, Types of data analytics (Descriptive, Diagnostic, Predictive, Prescriptive), Data preprocessing techniques for IoT data, Statistical analysis and visualization of IoT data

Module 5: Machine Learning for IoT Analytics: **[8L]**

Introduction to machine learning, Machine learning algorithms for IoT data analysis, Real-time predictive analytics for IoT, Anomaly detection and fault prediction in IoT systems

Module 6: Big Data Technologies for IoT : **[4L]**

Introduction to big data technologies, Distributed storage systems (HDFS, S3), Stream processing frameworks (Apache Kafka, Apache Flink), Batch processing frameworks (Apache Spark).

Module 7: Edge Computing for IoT: **[4L]**

Introduction to edge computing, Edge computing architectures, Edge analytics and decision-making, Integration of edge computing with cloud and IoT systems.

Module 8: Security and Privacy in IoT Cloud Processing: [**4L]**

Security challenges in IoT systems, Authentication and authorization mechanisms, Data encryption and privacy-preserving techniques, Security best practices for IoT cloud processing.

Module 9: Future Trends in IoT Cloud Processing **[6L]**

Emerging technologies and trends in IoT, Research challenges and opportunities in IoT cloud processing and analytics, Ethical considerations and societal impacts of IoT technologies.

Text/Reference Books:

1. Arshdeep Bahga and Vijay Madisetti, “Internet of Things – A Hands on Approach”, Universities Press, 2015.
2. Kevin Townsend, Carles Cufí, Akiba and Robert Davidson, "Getting Started with Bluetooth Low Energy”, O'Reilly.
3. Madhur Bhargava, “IoT Projects with Bluetooth Low Energy”, Packt Publishing, August 2017.
4. Robin Heydon, “Bluetooth Low Energy: The Developer's Handbook”, Pearson, October 2012.
5. Kumar Saurabh, “Cloud Computing”, Wiley India, 1st Edition, 2016.

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS7002F			
Course Title	Data Science For Iot			
Category	ENGG Major			
LTP & Credits	L	T	P	Credits
	3	0	0	3
Total Contact Hours	36			
Pre-requisites	None/If Any			

Learning Objective:

This syllabus covers the fundamental concepts, techniques, and applications of Data Science in the context of Internet of Things (IoT) systems. It provides students with both theoretical knowledge and practical skills necessary to analyze, interpret, and derive insights from IoT data using Data Science

techniques.

Course Outcome:

CO1: Understand the fundamental concepts and principles of Data Science and its relevance to Internet of Things (IoT) systems.
CO2: Demonstrate knowledge of the architecture, components, and technologies involved in IoT environments.
CO3: Acquire skills in data acquisition, preprocessing, and integration from heterogeneous sources in IoT setups.
CO4: Apply various data storage solutions and management techniques for handling large-scale IoT datasets effectively.
CO5: Analyze IoT data using statistical methods, machine learning algorithms, and exploratory data analysis techniques

Course Content:

Module 1: Introduction to Data Science and IoT [6L]

Overview of Data Science: Definition, scope, and applications,Introduction to Internet of Things (IoT): Concepts, architecture, and components,Intersection of Data Science and IoT: Opportunities and challenges, Use cases and real-world applications of Data Science in IoT

Module 2: Data Acquisition and Preprocessing for IoT [7L]

Data Collection in IoT: Sensors, devices, and data sources, Data Cleaning and Preprocessing: Dealing with missing values, noise, and outliers, Time-Series Data Handling: Techniques for analyzing time-series data from IoT sensors, Data Integration and Fusion: Merging data from heterogeneous sources in IoT environments

Module 3: Data Storage and Management for IoT [8L}

IoT Data Storage Architectures: Relational databases, NoSQL databases, and time-series databases, Scalable Data Storage Solutions for IoT: Big Data platforms and cloud-based storage, Data Security and Privacy in IoT: Encryption, access control, and compliance considerations, Stream Processing and Real-time Analytics: Techniques for processing and analyzing streaming data from IoT devices

Module 4: Data Analysis and Visualization for IoT [5L]

Exploratory Data Analysis (EDA) techniques for IoT datasets, Statistical Analysis: Descriptive statistics, hypothesis testing, and correlation analysis, Machine Learning for IoT: Regression, classification, clustering , and anomaly detection, Visualization Techniques for IoT Data: Dashboard design, interactive visualizations, and storytelling with data

Module 5: IoT Analytics and Decision Support [4L]

Predictive Analytics for IoT: Forecasting techniques and predictive modeling, Prescriptive Analytics: Recommender systems and decision support algorithms, Edge Analytics in IoT: Moving analytics closer to IoT devices for real-time insights, Optimization and Control in IoT: Optimization algorithms for resource allocation and control systems

Module 6: Case Studies and Projects [6L]

Case studies on the application of Data Science techniques in IoT across various domains (e.g., healthcare, smart cities, industrial IoT), Project Work: Students will work on a hands-on project where they apply Data Science techniques to analyze and derive insights from real IoT datasets, Presentation and Demonstration:

Students will present their project findings and demonstrate the practical applications of Data Science in IoT.

Text/Reference Books:

1. Kai Hwang, Min Chen, Big-Data Analytics for Cloud, IoT and Cognitive Computing, Wiley.
2. Hwaiyu Geng, Internet of Things and Data Analytics Handbook, Wiley.
3. John Soldatos, Building Blocks for IoT Analytics Internet-of-Things Analytics, River Publishers Gerardus Blokdyk.
4. IoT Analytics A Complete Guide, 5starcooks.

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS 7002G			
Course Title	Artificial Intelligence			
Category	Major			
LTP& Credits	L	T	P	Credits
	3	0	0	3
Total Contact Hours	36			
Pre-requisites	a)Design and Analysis of Algorithms			

Learning Objective:

In this course, students will learn the basic concepts, theories, and techniques of artificial intelligence (AI) and understand the application of machine learning/AI algorithms in different fields of Computer Engineering.

Course Outcome:

CO1: Explain the basic concept of Artificial Intelligence and its applications.
CO2: Classify and analyze various AI tools and techniques.
CO3: Learn and evaluate various AI algorithms.
CO4: Apply the basic understanding of artificial intelligence in real-world applications.
CO5: Evaluate and assess the performance, effectiveness, and ethical implications of artificial Intelligence systems in various real-world applications.

Course Content:

Module 1: Introduction to Artificial Intelligence [6L]

- Overview: foundations, scope, problems, and approaches of AI.
- Intelligent agents: re-active, deliberative, goal-driven, utility-driven, and learning agents.

Module 2: AI Techniques [8L]

- Artificial Intelligence programming techniques.
- Problem-solving through Search: forward and backward, state-space, blind, heuristic, problem-reduction, A,
- A*, AO*, min-max, constraint propagation, neural, stochastic, and evolutionary search algorithms, sample applications.

Module 3:
Planning and Representation in AI [7L]

- Planning: planning as search, partial order planning, construction and use of planning graphs.
- Representing and Reasoning with Uncertain Knowledge: probability, connection to logic, independence,
- Bayes rule, Bayesian networks, probabilistic inference, sample applications.

Module 4: Decision Making [8L]

- Decision-Making: basics of utility theory, decision theory, sequential decision problems, elementary game theory, sample applications.

Module 5: Knowledge Acquisition [7L]

- Machine Learning and Knowledge Acquisition: learning from memorization, examples, explanation, and exploration.
- Learning nearest neighbor, naive Bayes, and decision tree classifiers, Q-learning for learning action policies, applications.

Text/Reference Books:

1. S.Russell and P.Norvig, "Artificial intelligence: A Modern Approach", Prentice Hall.
2. N.J.Nilsson,"ArtificialIntelligence:ANewSynthesis",Morgan-Kaufmann,1998.
3. J.Pearl,"Heuristics:IntelligentSearchStrategiesforComputerProblemSolving",Addison-WesleyPublishingCompany.
4. B.A.Heule, M.VanMaaren and H. Walsh, "The Handbook of Satisfiability", IOS Press.

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS7002H
Course Title	Fog Computing
Category	ENGG Major

LTP & Credits	L	T	P	Credits
	3	0	0	3
Total Contact Hours	36			
Pre-requisites	None/If Any			

Learning Objective:

This syllabus covers the fundamental aspects, architectures, programming models, data management, communication protocols, applications, challenges, and future directions in the field of Fog Computing.

Course Outcome:

CO1: A comprehensive understanding of Fog Computing principles, architectures, and components, including its comparison with Cloud Computing and Edge Computing.
CO2: Identify and evaluate the hardware and software requirements for Fog Computing infrastructure, including virtualization techniques and networking setups.
CO3: Design and implement effective task offloading strategies, resource management techniques, and load balancing algorithms for Fog Computing environments.
CO4: The necessary programming skills and knowledge of middleware and APIs to develop applications tailored for Fog Computing platforms.
CO5: Manage data effectively in Fog Computing systems, including storage, retrieval, processing, analytics, replication, consistency, security, and privacy considerations.

Course Content:

Module 1: Introduction to Fog Computing [5L]
- Overview of Fog Computing
- Comparison with Cloud Computing and Edge Computing
- Motivation and Applications of Fog Computing
- Architecture and Components of Fog Computing

Module 2: Fog Computing Infrastructure[5L]
- Hardware Requirements for Fog Computing
- Software Requirements and Platforms
- Virtualization Techniques in Fog Computing
- Networking Infrastructure for Fog Computing

Module 3: Fog Computing Models[4L]
- Task Offloading Strategies
- Resource Management and Allocation
- Load Balancing Techniques in Fog Computing
- Security and Privacy Considerations

Module 4: Programming Models for Fog Computing [4L]
- Fog Programming Paradigms
- Developing Applications for Fog Computing
- Middleware and APIs for Fog Computing
- Case Studies and Examples

Module 5: Data Management in Fog Computing [6L]
- Data Storage and Retrieval in Fog Computing
- Data Processing and Analytics at the Edge
- Data Replication and Consistency Models
- Data Security and Privacy in Fog Computing

Module 6: Communication Protocols in Fog Computing[4L]
- Communication Models in Fog Computing
- IoT Protocols and Standards
- Fog-to-Cloud Communication Protocols

- QoS and Reliability in Fog Communication

Module 7: Fog Computing Applications[4L]

- Smart Cities and Urban Computing
- Industrial IoT and Industry 4.0
- Healthcare Applications of Fog Computing
- Transportation and Logistics Applications

Module 8: Challenges and Future Directions[4L]

- Scalability and Performance Challenges
- Interoperability and Standardization
- Energy Efficiency in Fog Computing
- Emerging Trends and Future Directions in Fog Computing

Text/References Books:

1. "Fog Computing: Concepts, Frameworks and Applications" by Rajkumar Buyya, Amir Vahid Dastjerdi, and Satish Narayana Srirama.
2. "Fog Computing: A Practical Guide to Container Orchestration and Microservices in Fog Environments" by K.K. Singh, Yogesh Raheja, and Arun Sharma
3. "Fog Computing in the Internet of Things: Intelligence at the Edge" by Zaigham Mahmood and Rajkumar Buyya.
4. "Fog Computing: Principles, Architectures, and Applications" by Moussa Ayyash and Majd F. Sakr

CO/PO mapping:

Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS7003G			
Course Title	Block chain Technology			
Category	Major ENGG			
LTP& Credits	L	T	P	Credits
	3		0	3
Total Contact Hours	36			
Pre-requisites				

Learning Objectives:

Blockchain is an emerging technology platform for developing decentralized applications and data storage, extending beyond its role as the technology underlying crypto currencies. The basic tenet of this platform is to create a distributed and replicated ledger of events, transactions, and data generated through various IT processes with strong cryptographic guarantees of tamper resistance, immutability, and verifiability. Public block chain platforms allow us to guarantee these properties with overwhelming probabilities even when untrusted users are participants of distributed applications with the ability to transact on the platform. Although block chain technology has become popularly known because of its use in implementing crypto currencies such as Bitcoin, Ethereum, etc.

Course Outcomes:

CO 1: Describe the basic concepts and technology used for blockchain.

CO 2: Describe the primitives of distributed computing and cryptography related to blockchain.

CO 3: Illustrate the concepts of Bitcoin and their usage.

CO 4: Implement Ethereum blockchain contracts.

CO 5: Apply security features in blockchain technologies.

Course Contents:

Module 1: Introduction [6L]

- Need for Distributed Record Keeping.
- Modeling faults and adversaries.
- Byzantine Generals problem.
- Consensus algorithms and their scalability problems.
- Why Nakamoto Came up with Blockchain-based cryptocurrency?
- Technologies Borrowed in Blockchain – hash pointers, consensus, Byzantine fault-tolerant distributed computing, digital cash, etc.

Module 2: Basic Distributed Computing [4L]

- Atomic Broadcast, Consensus, Byzantine Models of fault tolerance.

Module 3: Basic Crypto primitives [6L]

- Hash functions, Puzzle-friendly Hash, Collision-resistant hash, digital signatures, public key crypto, verifiable random functions, Zero-knowledge systems.

Module 4: Blockchain 1.0 [9L]

- Bitcoin blockchain, the challenges, and solutions.
- Proof of work, Proof of stake, alternatives to Bitcoin consensus.
- Bitcoin scripting language and their use.

Module 5: Blockchain 2.0 [9L]

- Ethereum and Smart Contracts.
- The Turing Completeness of Smart Contract Languages and verification challenges.
- Using smart contracts to enforce legal contracts.
- Comparing Bitcoin scripting vs. Ethereum Smart Contracts.

Module 5: Blockchain 3.0 [6L]

- Hyperledger Fabric, the plug and play platform, and mechanisms in permission blockchain.

Module 6: Privacy, Security issues in Blockchain [8L]

- Pseudo-anonymity vs. anonymity.
- Zcash and Zk-SNARKS for anonymity preservation.
- Attacks on Blockchains – such as Sybil attacks, selfish mining, 51% attacks.
- Advent of Algorand and Sharding-based consensus algorithms to prevent these.

Text / Reference Books: Josh Thompson, 'Blockchain: The Blockchain for Beginnings, Guide to Blockchain Technology and Blockchain Programming', Create Space Independent Publishing Platform, 2017.

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS7003H			
Course Title	Ad Hoc And Sensor Networks			
Category	Major ENGG			
LTP& Credits	L	T	P	Credits
	3		0	3
Total Contact Hours	36			
Pre-requisites				

Learning Objectives:

- Understand the concept of ad hoc networks and their characteristics.
- Explain the architecture and routing algorithms for wireless sensor networks.
- Identify the challenges and solutions in implementing ad hoc and sensor networks.
- Implement routing algorithms based on network and user requirements.
- Familiarize with physical and MAC layer protocols and their issues.
- Demonstrate knowledge of transport layer and security issues in ad hoc and sensor networks.
- Develop basic modules using the operating systems used in wireless sensor networks.

Course Outcomes:

- CO1: Construct ad hoc networks and wireless sensor networks.
- CO2: Implement suitable routing algorithms based on network and user requirements.
- CO3: Identify appropriate physical and MAC layer protocols and their issues.
- CO4: Demonstrate knowledge of transport layer and security issues in ad hoc and sensor networks.
- CO5: Be familiar with the operating systems used in wireless sensor networks and build basic modules.

Course Contents:

Module 1: Introduction [7L]

- Characteristics of MANETs
- Application and challenges of MANETs
- Topology-based vs position-based approaches
- Topology-based routing protocols
- Position-based routing protocols
- Other routing protocols

Module 2: Data Transmission in MANETs [6L]

- Broadcast storm multicasting
- Geocasting
- TCP protocol overview
- TCP and MANETs
- Solutions for TCP over Ad Hoc

Module 3: Basics of Wireless Sensors [8L]

- Introduction to wireless sensors and applications
- Design issues and energy consumption
- Clustering of sensors
- Applications of wireless sensor networks
- Data retrieval in sensor networks

Module 4: Security and Sensor Network Platforms [7L]

- Security in ad hoc wireless networks
- Key management and secure routing cooperation in MANETs
- Intrusion detection systems
- Sensor network hardware and programming challenges
- Node-level software platforms

Module 5: Operating Systems and Simulators [6L]

- Operating system: TinyOS
- Imperative language: nesC
- Dataflow style language: TinyGALS
- Node-level simulators: ns-2 and its sensor networks extension, TOSSIM

Text / Reference Books:

1. C. Siva Ram Murthy and B. S. Manoj, "Ad Hoc Wireless Networks Architectures and Protocols", Prentice Hall, PTR, 2004.
2. Holger Karl, Andreas Willig, "Protocol and Architecture for Wireless Sensor Networks", John Wiley publication, Jan 2006.
3. Ad Hoc and Sensor Networks, Carlos Corderio, Dharma P. Aggarwal, World Scientific Publication/ Cambridge University Press, March 2006.
4. Charles E. Perkins, "Ad Hoc Networking", Addison Wesley, 2000.
5. Wireless Sensor Networks: An Information Processing Approach, Feng Zhao, Leonidas Guibas, Elsevier Science imprint, Morgan Kaufmann publishers, 2005,rp2000

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS7003I			
Course Title	Augmented Reality And Virtual Reality			
Category	ENGG Major			
LTP & Credits	L	T	P	Credits
	3	0	0	3
Total Contact Hours	36			
Pre-requisites	None/If Any			

Learning Objective:

The course covers a comprehensive range of topics to provide students with a solid understanding of both the theoretical and practical aspects of Augmented Reality and Virtual Reality.

Course Outcome:

CO1: Understand the fundamental concepts, principles, and technologies underlying Augmented Reality (AR) and Virtual Reality (VR).

CO2: Analyze and compare various AR and VR systems, including their architectures, components, and functionalities.

CO3: Apply the principles of computer graphics to design and develop immersive AR and VR experiences.

CO4: Implement AR applications using marker-based and markerless tracking techniques, registration, and calibration methods.

CO5: Implement VR applications using head-mounted displays (HMDs), input devices, and immersive interaction techniques.

Course Content:

Module 1: Introduction to Augmented Reality and Virtual Reality

- Overview of Augmented Reality (AR) and Virtual Reality (VR)
- Historical development and evolution
- Key concepts and definitions
- Applications and use cases in various fields
- Comparison between AR and VR technologies

Module 2: Fundamentals of Computer Graphics

- Introduction to computer graphics
- 2D and 3D graphics primitives
- Rendering techniques: rasterization vs. ray tracing
- Shading and lighting models
- Transformations and projections

Module 3: Augmented Reality Fundamentals

- Understanding the AR ecosystem
- Marker-based vs. markerless AR
- Tracking techniques: fiducial markers, image recognition, GPS, etc.
- Registration and calibration
- Interaction techniques in AR

Module 4: Virtual Reality Fundamentals

- Understanding the VR ecosystem
- Immersive technologies: head-mounted displays (HMDs), input devices
- Tracking techniques: inertial, optical, and hybrid tracking
- VR locomotion techniques
- Design principles for VR environments

Module 5: Augmented Reality Development

- AR development platforms and tools
- AR content creation: 3D modeling, animation, texturing
- Programming languages and frameworks for AR development (e.g., ARKit, ARCore)
- Integration with other technologies (e.g., IoT, AI)

Module 6: Virtual Reality Development

- VR development platforms and tools
- VR content creation: 3D modeling, animation, texturing for VR environments
- Programming languages and frameworks for VR development (e.g., Unity, Unreal Engine)

- Optimization techniques for VR performance

Module 7: Advanced Topics in Augmented Reality

- Advanced AR tracking techniques
- Simultaneous Localization and Mapping (SLAM)
- Augmented Reality Cloud services
- Augmented Reality for specialized applications (e.g., medical, education, retail)

Module 8: Advanced Topics in Virtual Reality

- Advanced VR interaction techniques
- Multi-user VR environments
- VR sickness mitigation techniques
- VR for training and simulation applications

Module 9: Ethical and Social Implications

- Ethical considerations in AR and VR development and usage
- Social impact and implications on society
- Privacy and security concerns
- Regulatory frameworks and guidelines

Text/References Books:

1. "Augmented Reality: Principles and Practice" by Dieter Schmalstieg and Tobias Hollerer, Addison-Wesley Professional
2. "Virtual Reality: Concepts and Technologies" by Philippe Fuchs, Pascal Guitton, and Pierre Boulanger, CRC Press

CO/PO mapping

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS7004D			
Course Title	Advanced Algorithms			
Category	CS			
LTP& Credits	L	T	P	Credits
	3	0	0	3
Total Contact Hours	36			
Pre-requisites	Data structures			

Learning Objectives:

Introduce students to the advanced methods of designing and analyzing algorithms. The student should be able to choose appropriate algorithms and use them for specific problems. To familiarize students with basic paradigms and data structures used to solve advanced algorithmic problems. Students should be able to understand different classes of problems concerning their computation difficulties. To introduce the students to recent developments in the area of algorithmic design.

Course Outcomes:

CO1: Analyze the complexity/performance of different algorithms.
CO2: Determine the appropriate data structure for solving a particular set of problems.
CO3: Categorize different problems into various classes according to their complexity.
CO4: Students should have an insight into recent activities in the field of advanced data structures.
CO5: Apply advanced algorithms to solve real-world problems effectively.

Course Contents:

Module 1: [6L]

Sorting: Review of various sorting algorithms, topological sorting. Graph: Definitions and Elementary Algorithms: Shortest path by BFS, shortest path in edge-weighted case (Dijkstra's), depth-first search, and computation of strongly connected components, emphasis on correctness proof of the algorithm and time /space analysis, example of amortized analysis.

Module 2: [8L]

Matroids: Introduction to greedy paradigm, algorithm to compute a maximum weight maximal independent set. Application to MST. Graph Matching: Algorithm to compute maximum matching. Characterization of maximum matching by augmenting paths, Edmond's Blossom algorithm to compute augmenting path.

Module 3: [8L]

Flow-Networks: Maxflow-mincut theorem, Ford-Fulkerson Method to compute maximum flow, Edmond-Karp maximum-flow algorithm. Matrix Computations: Strassen's algorithm and introduction to divide and conquer paradigm, inverse of a triangular matrix, relation between the time complexities of basic matrix operations, LUP-decomposition.

Module 4: [8L]

Shortest Path in Graphs: Floyd-Warshall algorithm and introduction to dynamic programming paradigm. Modulo Representation of integers/polynomials: Chinese Remainder Theorem, Conversion between base-representation and modulo-representation. Discrete Fourier Transform (DFT).

Module 5: [6L]

Linear Programming: Geometry of the feasibility region and Simplex algorithm. NP-completeness: Examples, proof of NP-hardness and NP-completeness. One or more of the following topics based on time and interest: Approximation algorithms, Randomized Algorithms, Interior Point Method, Advanced Number.

Text/Reference Books:

1. "Introduction to Algorithms" by Cormen, Leiserson, Rivest, Stein.
2. "The Design and Analysis of Computer Algorithms" by Aho, Hopcroft, Ullman.
3. "Algorithm Design" by Kleinberg and Tardos.

CO-PO Mapping

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS7004E			
Course Title	Neural Networks and Deep Learning			
Category	Major ENGG			
LTP & Credits	L	T	P	Credits
	3	0	0	3
Total Contact Hours	36			
Pre-requisites				

Learning Objective:

Neural networks enable computers to make intelligent decisions with minimal human intervention by learning and modeling complex, nonlinear relationships between input and output data. Deep learning encompasses a wide range of highly effective techniques in machine learning. The course aims to familiarize students with this area and provide them with the knowledge to apply these techniques to solve real-world problems.

Course Outcome:

CO1: Understand the mathematics behind the functioning of artificial neural networks.

CO2: Analyze datasets to design neural network-based solutions.

CO3: Design and implement deep learning models for signal/image processing applications.

CO4: Design and deploy simple TensorFlow-based deep learning solutions for classification problems.

CO5: Apply deep learning techniques to solve complex real-world problems.

Course Content:

Module 1: Introduction (4L)

- Various paradigms of learning problems.
- Perspectives and issues in deep learning framework.
- Review of fundamental learning techniques.

Module 2: Feed Forward Neural Network (6L)

- Artificial Neural Network.
- Activation functions.
- Multi-layer neural network.
- Cardinality, operations, and properties of fuzzy relations.

Module 3: Training Neural Network (6L)

- Risk minimization.
- Loss function.
- Backpropagation.
- Regularization.
- Model selection and optimization.

Module 4: Conditional Random Fields (8L)

- Linear chain.
- Partition function.
- Markov network.
- Belief propagation.
- Training CRFs.
- Hidden Markov Model.
- Entropy.

Module 5: Deep Learning (6L)

- Deep Feed Forward network.
- Regularizations.
- Training deep models.
- Dropouts.
- Convolution Neural Network.
- Recurrent Neural Network.
- Deep Belief Network.

Module 6: Deep Learning Research (6L)

- Object recognition.
- Sparse coding.
- Computer vision.
- Natural language.

Text/Reference Books:

1. Goodfellow, I., Bengio, Y., and Courville, A. "Deep Learning," MIT Press, 2016.
2. Bishop, C. M. "Pattern Recognition and Machine Learning," Springer, 2006.
3. Yegnanarayana, B. "Artificial Neural Networks," PHI Learning Pvt. Ltd, 2009.

4. Golub, G. H., and Van Loan, C. F. "Matrix Computations," JHU Press, 2013.
5. Kumar, Satish. "Neural Networks: A Classroom Approach," Tata McGraw-Hill Education, 2004.
6. Chopra, Rajiv. "Deep Learning," Khanna Publishing House, 2018.

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS7004F			
Course Title	Learning Optimization Technique			
Category	Major ENGG			
LTP & Credits	L	T	P	Credits
	3	0	0	3
Total Contact Hours	36			
Pre-requisites				

Learning Objectives:

Introduce optimization techniques using both linear and non-linear programming, with a focus on convex optimization while covering some techniques for non-convex function optimization.

Provide students with a solid foundation in mathematical preliminaries, enabling them to frame engineering minima maxima problems in the framework of optimization problems.

Course Outcome:

CO1: Understand the mathematical preliminaries required for optimization techniques, including linear algebra, matrices, vector spaces, eigen analysis, probability theory, and multivariable calculus.

CO2: Apply linear programming techniques, including the simplex method, duality, and Karmarkar's method, to solve optimization problems.

CO3: Apply unconstrained optimization methods, such as conjugate direction and quasi-Newton methods, gradient-based methods, and one-dimensional search methods, to solve optimization problems.

CO4: Apply constrained optimization techniques, including Lagrange theorem, FONC, SONC, and SOSC conditions, to solve optimization problems with constraints.

CO5: Understand and apply projection methods, KKT conditions, and models for non-linear constrained optimization problems.

Course Content:

Module 1: Mathematical Preliminaries [8 L]

- Linear algebra and matrices
- Vector spaces
- Eigen analysis
- Elements of probability theory
- Elementary multivariable calculus

Module 2: Linear Programming (6 Lectures)

- Simplex method
- Introduction to linear programming model
- Duality
- Karmarkar's method

Module 3: Unconstrained Optimization (6 Lectures)

- Conjugate direction and quasi-Newton methods
- Gradient-based methods
- One-dimensional search methods

Module 4: Constrained Optimization (8 Lectures)

- Lagrange theorem
- FONC, SONC, and SOSC conditions

Module 5: Non-linear Constrained Optimization (8 Lectures)

- Projection methods
- KKT conditions
- Non-linear constrained optimization models
- Non-linear problems

Text/Reference Books:

1. An Introduction to Optimization by Edwin P K Chong, Stainslaw Zak
2. Nonlinear Programming by Dimitri Bertsekas

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YBB7001			
Course Title	Human Resource Development and Organizational Behavior			
Category	Minor HUM			
LTP & Credits	**L**	**T**	**P**	Credits
	2	0	0	1
Total Contact Hours	36			
Pre-requisites				

Learning Objective:

To provide students with an understanding of the basic principles of organizational behavior, equipping them with managerial skills and the necessary knowledge in human resource management.

Course Outcome:

CO1: Define and explain the basic concepts of organizational behavior and motivation.
CO2: Explain essential concepts of organizational conflicts, conflict resolution through negotiation, change management, and organizational development.
CO3: Familiarize students with various aspects of HR, enabling them to deal effectively with people resourcing, talent management, and HR functions in an organization.
CO4: Understand the concepts of HRD, its role, and importance in the success of an organization.
CO5: Develop an understanding of compensation management and industrial relations.

Course Content:

Module 1: Introduction to Organizational Behavior (8L)

- Meaning and scope of organizational behavior
- Challenges and Opportunities
- Foundations of Individual behavior
- Motivation Theories (Maslow, ERG, Douglas McGregor two-factor theory)
- Group dynamics
- Leadership styles

Module 2: Organizational Conflict and Change Management (7L)

- Causes and consequences of organizational conflict
- Conflict and negotiation
- Organizational change
- Change management process
- Resistance to change

- Flexibility and crisis management
- Organizational Development: Concept and significance

Module 3: Human Resource Management (6L)

- Meaning, definition, and functions of HRM
- Job Analysis and Job Design
- Human Resource Planning
- Recruitment and Selection
- Sources of Recruitment
- Selection process, Placement, and Induction

Module 4: Human Resource Development (7L)

- Concepts of Human Resource Development
- Training and Development
- Methods of training
- Importance of Performance Appraisal
- Traditional and modern methods of performance appraisal
- Job Evaluation
- Methods of Job Evaluation
- Wage and Salary Administration

Module 5: Compensation Management and Industrial Relations (8L)

- Concepts and Principles of Compensation
- Influencing Factors
- Emerging Trends in Compensation
- Methods of Payment
- Incentives and Rewards
- Managing Industrial Relations
- Emerging trends and practices in human resource management

Text/Reference Books:

1. K. Aswathappa, Organizational Behaviour, 12th edition, Himalaya, 2016
2. Edwin B. Flippo, Personnel Management, 6th edition, TMH, 2013
3. P. Subba Rao, Management & Organizational Behavior, 2nd edition, Himalaya, 2014
4. C.B. Mamoria& VSP Rao, Personnel Management, 20th edition, Himalaya, 2015
5. Stephen P. Robins, Organisational Behaviour, 11th edition, PHI Learning / Pearson Education, 2008

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS7106			
Course Title	Iot Cloud Processing And Analytics Laboratory			
Category	ENGG Major			
LTP & Credits	L	T	P	Credits
	0	0	3	1.5
Total Contact Hours	36			
Pre-requisites	None/If Any			

Learning Objective:

This course covers fundamental concepts, practical applications, and advanced topics related to IoT cloud processing and analytics, providing students with a comprehensive understanding of the subject.

Course Outcome:

CO1: Understand the fundamental concepts of IoT architecture, including sensors, communication protocols, and data transmission mechanisms.
CO2: Develop skills in data analytics for IoT, including data preprocessing, statistical analysis, and visualization of IoT data.
CO3: Learn about big data technologies and frameworks for handling large volumes of IoT data, both in batch and streaming modes.
CO4: Gain insights into security and privacy challenges in IoT cloud processing and acquire skills in implementing security mechanisms and best practices.
CO5: Identify and discuss emerging trends and future directions in IoT cloud processing, including ethical considerations and societal impacts.

Course Content:

1. Lab 1: Introduction to IoT and Cloud Computing

Overview of IoT concepts and applications
Introduction to cloud computing and its relevance to IoT
Setting up cloud computing environments (e.g., AWS, Azure, Google Cloud)

2. Lab 2: IoT Devices and Sensors

Understanding various IoT devices and sensors
Hands-on experience with setting up and programming IoT devices (e.g., Arduino, Raspberry Pi)
Data collection from sensors and transmitting to the cloud

3. Lab 3: Data Communication Protocols for IoT

Overview of communication protocols (e.g., MQTT, HTTP, CoAP)
Hands-on exercises on implementing different protocols for IoT communication

4. Lab 4: Cloud Data Storage for IoT

Introduction to cloud-based databases (e.g., AWS DynamoDB, Google Cloud Firestore)
Hands-on experience with storing IoT data in the cloud

5. Lab 5: IoT Data Processing

Basics of data processing techniques for IoT data
Hands-on exercises with data preprocessing and cleaning

6. Lab 6: Real-time Data Processing and Stream Analytics

Introduction to stream processing frameworks (e.g., Apache Kafka, AWS Kinesis)
Hands-on experience with real-time data processing and analytics

7. Lab 7: Edge Computing for IoT

Understanding the concept of edge computing and its importance in IoT
Hands-on exercises with deploying edge computing solutions

8. Lab 8: IoT Security and Privacy

Overview of IoT security challenges and best practices
Hands-on experience with implementing security measures in IoT systems

9. Lab 9: Machine Learning for IoT Analytics

Introduction to machine learning algorithms for IoT analytics
Hands-on exercises with building predictive models using IoT data

10. Lab 10: Data Visualization and Dash-boarding

Overview of data visualization techniques
Hands-on experience with creating dashboards for IoT data analysis

Text/Reference Books:

1. Arshdeep Bahga and Vijay Madisetti, “Internet of Things – A Hands on Approach”, Universities Press, 2015.
2. Kevin, Townsend, Carles, Cufí, Akiba and Robert Davidson, "Getting Started with Bluetooth Low Energy” O'Reilly.
3. Madhur Bhargava “IoT Projects with Bluetooth Low Energy, Packt Publishing, August 2017.
4. Robin Heydon,” Bluetooth Low Energy: The Developer's Handbook”, Pearson, October 2012
Kumar Saurabh,” Cloud Computing”, Wiley India, 1st Edition, 2016.

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS7102F			
Course Title	Data Science Laboratory			
Category	Major			
LTP & Credits	L	T	P	Credits
	0	0	3	1.5
Total Contact Hours	36			
Pre-requisites				

Learning Objective:

Apply quantitative modelling and data analysis techniques to the solution of real-world business problems, communicate findings, and effectively present results using data visualization techniques. To demonstrate knowledge of statistical data analysis techniques utilized in business decision making. To apply principles of Data Science to analyze the business problems.

Course Outcome:

CO1: Students will become proficient in the statistical analysis of data and the use of computation tools for data analysis.

CO2: Students will apply statistical and computational tools to applied problems.

CO3: Students will clearly communicate the results in both written reports and oral presentations.

CO4: Students will understand the importance of proper data management, documentation of work to allow reproducibility of results

CO5: Assess the ethical considerations of a data science project.

Course Content:

List of Experiments:

1. Interactive commands in Python, data operations, simple programs for writing into files and reading from files. Data file manipulations programs.
2. Familiarization with IDE in Python
3. Writing programs for standard algorithms of sorting and searching in Python.
4. Plotting the data using X-Y graph, Bar- chart, and using other plotting techniques.
5. Write programs to perform exploratory data analysis: variance, standard derivation, summarization, distribution, and statistical inference.
6. Plotting the various distributions for given data sets.
7. Classifying and presentation of data using support vector machine.
8. Write programs for k-means clustering and presentation for given data sets.
9. Write programs on graphs of social networks for community detection
10. Write programs for analysis of graphs to find centrality and page-rank.

Text/Reference Books:

1.Kai Hwang, Min Chen, Big-Data Analytics for Cloud, IoT and Cognitive Computing, Wiley.
2. Hwaiyu Geng, Internet of Things and Data Analytics Handbook, Wiley.
3. John Soldatos, Building Blocks for IoT Analytics Internet-of-Things Analytics, River Publishers Gerardus Blokdyk.
4. IoT Analytics A Complete Guide, 5starcooks.

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS7102G				
Course Title	Artificial Intelligence Laboratory				
Category	Major				
LTP& Credits	L	T	P	Credits	
	0	0	3	1.5	
Total Contact Hours	36				
Pre-requisites	b) Data Structures and Algorithms c) Database Management System d) Programming Practices II				

Learning Objective:

In this course, students will learn about the basic principles, techniques, and applications of Artificial Intelligence (AI) and Machine Learning (ML) towards problem-solving, inference, perception, knowledge representation, and puzzles design.

Course Outcome:

CO1: Explain the working principles of PROLOG /LISP and apply LIST structure of PROLOG.

CO2: Apply the principles of reasoning and inference to real-world problems and design programs to solve various puzzles.

CO3: Design simple algorithms for data classification in Python /R and test them with benchmark datasets.

CO4: Design simple algorithms for data clusteri in Python / R and test them with benchmark datasets.

CO5: Analyze and evaluate algorithms for estimation / prediction using regression.

Suggestive List of Experiments: This laboratory course will familiarize students with PROLOG/LISP language through four modules:

Module 1: Introduction to PROLOG

- Familiarization with facts & rules using a simple family tree.
- Execution of goals in PROLOG and simple queries on the family tree.
- Formation of recursive definitions and handling problems with numbers.
- Implementation of Graph Search algorithms like DFS, BFS.

- Implementation of well-known puzzles like 8-queens problem, Towers-of-Hanoi problem, etc.

Module 2: Implementation of Classifiers

- Implementing classifiers such as KNN, Naive Bayes, Decision Tree, SVM, Perceptron, Multi-Layer Perceptron, Random Forest on Python/R platform.
- Testing classifiers on benchmark datasets (Kaggle / UCI Machine Learning).
- Familiarization with ML Tools: Excel, WEKA, R, Python for classification.

Module 3: Implementation of Data Clustering Algorithms

- Implementing clustering algorithms like K-Means, DBSCAN, Hierarchical (AGNES/-DIVISIVE) on Python/R platform.
- Testing clustering algorithms on benchmark datasets (Kaggle/UCI Machine Learning).
- Familiarization with ML Tools: Excel, WEKA, R, Python for clustering.

Module 4: Implementation of Regression

- Implementing linear and non-linear Regression (single and multiple variables), Logistic Regression for prediction tasks.

Text/Reference Books:

1. Bratko, "Prolog programming for artificial intelligence", Pearson education.
2. S. Kaushi, "Logic and Prolog Programming", New Age International Publishers.
3. B. Lantz, "Machine learning with R", PACKT Publishing.
4. C. M. Andreas and S. Guido, "Introduction to Machine Learning with Python: A Guide for Data Scientists", O'ReillyMedia.

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS7102H			
Course Title	Cloud Computing Laboratory			
Category	Major			
LTP& Credits	L	T	P	Credits
	0	0	3	1.5
Total Contact Hours	36			

Pre-requisites	Distributed Computing, Operating Systems, Grid and Cloud Computing

Learning Objective:

Be exposed to tool kits for grid and cloud environment. Be familiar with developing web services/ Applications in grid framework. Learn to run virtual machines of different configuration.

Course Outcome:

CO1: Use the grid and cloud tool kits.
CO2: Design and implement applications on the Grid.
CO3: Design and Implement applications on the Cloud.
CO4: Connect Multiple System Using Zonal Server and JVishwa
CO5: Implement the Theorem using Naive Bayes Approach.

Course Contents:

List of Experiments :

1. Install Virtualbox/VMware Workstation with different flavours of linux or windows OS on top of windows7 or 8.

2. Install a C compiler in the virtual machine created using virtual box and execute Simple Programs.
3. Install Google App Engine. Create hello world app and other simple web applications using python/java.

4. Use GAE launcher to launch the web applications.

5. Simulate a cloud scenario using CloudSim and run a scheduling algorithm that is not present in CloudSim.

6. Find a procedure to transfer the files from one virtual machine to another virtual machine.

7. Find a procedure to launch virtual machine using trystack (Online Openstack Demo Version)

8. Install Hadoop single node cluster and run simple applications like wordcount.

Text/Reference Books:

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Semester 8 Curriculum & Syllabus

Course Code	YCS8001G			
Course Title	Sensor and Devices			
Category	Major ENGG			
LTP & Credits	L	T	P	Credits
	3	1		4
Total Contact Hours	36			
Pre-requisites	None/ If Any			

Learning Objective:

Understanding of IoT value chain structure (device, data cloud), application areas and technologies involved. Understand IoT sensors and technological challenges faced by IoT devices, with a focus on wireless, energy, power, and sensing modules Market forecast for IoT devices with a focus on sensors .Explore and learn about Internet of Things with the help of preparing projects designed for Raspberry Pi

Course Outcome**:**

CO1: To introduce the terminology, technology and its applications.
CO2: To introduce the concept of M2M (machine to machine) with necessary protocols
CO3:.To introduce the Python Scripting Language which is used in many IoT devices
CO4: To introduce the Raspberry PI platform, that is widely used in IoT applications
CO5: To introduce the implementation of web-based services on IoT devices

Course Content:

Module 1: [7L]

Introduction to Signals and systems – Brief introduction Introduction to Internet of Things – Definition and Characteristics of IoT, Sensors, Actuators, Physical Design of loT – loT Protocols, loT communication models, loT Communication APIs, loT enabled Technologies – Wireless Sensor Networks, Cloud Computing, Embedded Systems, loT Levels and Templates, Domain Specific loTs – Home, City, Environment, Energy, Agriculture and lndustry.

Module 2: [7L]

IoT and M2M- Software defined networks, network function virtualization, difference between SDN and

NFV for IoT, Basics of IoT System Management with NETCOZF, YANG- NETCONF, YANG, SNMP NETOPEE

Module 3: **[9L]**

Controlling Hardware- Connecting LED, Buzzer, Switching High Power devices with transistors, Controlling AC Power devices with Relays, Controlling servo motor, speed control of DC Motor, unipolar and bipolar Stepper motors. IoT Physical Devices and Endpoints – lntroduction to Arduino and Raspberry Pi – installation, interfaces (serial, SPl, l2C) Controlling Hardware – Connecting LED, Buzzer, Switching High Power devices with transistors, Controlling AC Power devices with Relays, Controlling servo motor, speed control of DC Motor, unipolar and bipolar Stepper motors

Module 4: **[7L]**

Sensors – Light sensor, temperature sensor with thermistor, voltage sensor, ADC and DAC, Temperature and Humidity Sensor DHT11, Motion Detection Sensors, Wireless Bluetooth Sensors, Level Sensors, USB Sensors, Embedded Sensors, Distance Measurement with ultrasound sensor

Module 5: **[6L]**

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

Text/Reference Books:

1. lnternet of Things – A Hands – on Approach, Arshdeep Bahga and Vijay Madisetti, Universities Press, 2015, lSBN: 9788173719547
2. Getting Started with Raspberry Pi, Matt Richardson and Shawn Wallace, O'Reilly (SPD), 2014, ISBN: 9789350239759
3. Raspberry Pi Cookbook, Software and Hardware Problems and solutions, Simon Monk, O'Reilly (SPD), 2016, ISBN 7989352133895

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS8001H			
Course Title	Data Analytics			
Category	Major ENGG			
LTP & Credits	L	T	P	Credits
	3	1		4
Total Contact Hours	36			
Pre-requisites	None/ If Any			

Learning Objective:

Introduce the Business intelligence concepts, techniques and models .Understand the modeling process behind business analytics. To analyze different data analysis tools and techniques

Course Outcome**:**

CO1: Understand the fundamental of Business Intelligence and to design a customized solution
CO2: Familiarize on the concepts, techniques and reporting methods of descriptive analytics and predictive analytics
CO3: Explore the methods used to analyse speech and text and implement optimized search engines
CO4: Design and implement Decision Support systems
CO5: Familiarize on the processes needed to develop, report, and analyse business data

Course Content:

Module 1: **[7L]**

Introduction To Data Analytics: Introduction to Data Analytics: Sources and nature of data, Classification of data (structured, semi-structured, unstructured),Characteristics of data, Introduction to Big Data platform, Need of data analytics, Evolution of analytic scalability, Analytic process and tools, Analysis vs reporting, Modern data analytic tools, Applications of data analytics. Data Analytics Lifecycle: Need, key roles for successful analytic projects, Various phases of data analytics lifecycle – discovery, Data preparation, model planning, model building,

Communicating results, operationalization

Module 2: **[7L]**

Data Analysis: Regression modeling, Multivariate analysis, Bayesian modeling inference and Bayesian networks, Support vector and kernel methods, Analysis of time series: linear systems analysis & nonlinear dynamics, Rule induction, neural networks: learning and generalization, Competitive learning, Principal component analysis and neural networks, Fuzzy logic: extracting fuzzy models from data, Fuzzy decision trees, Stochastic search methods.

Module 3: **[9L]**

Mining DataStream: Introduction to streams concepts, Stream data model and architecture, Stream computing, Sampling data in a stream, Filtering streams, Counting distinct elements in a stream, Estimating moments, Counting oneness in a window, decaying window, Real-time Analytics Platform (RTAP) applications, Case studies – real time sentiment analysis,Stock market predictions.

Module 4: **[7L]**

Frequent Item sets & Clustering: Mining frequent item sets, Market based modelling, Apriori algorithm, Handling large data sets in main memory, Limited pas algorithm, Counting frequent item sets in a stream, Clustering techniques: hierarchical, K-means, Clustering high dimensional data, CLIQUE and ProCLUS, Frequent pattern based clustering methods, Clustering in non-euclidean space, Clustering for streams & parallelism.

Module 5: **[6L]**

Frame Works & Visualization: Frame Works and Visualization: MapReduce, Hadoop, Pig, Hive, HBase, MapR, Sharding, NoSQL Databases, S3, Hadoop Distributed File Systems, Visualization: visual data analysis techniques, interaction techniques, systems and applications.Introduction to R – R graphical user interfaces, data import and export, attribute and data types, descriptive statistics, exploratory data analysis, visualization before analysis, analytics for unstructured data.

Text/Reference Books:

1. Efraim Turban, Ramesh Sharda, Dursun Delen, “Business Intelligence and Analytics”, 10th Edition, Pearson , 2015.
2. S. Christian Albright, Wayne L. Winston, Business Analytics: Data Analysis & Decision Making, 6th Edition, CENGAGE INDIA , 2017
3. Rick Sherman, Business Intelligence Guidebook: From Data Integration to Analytics, Morgan Kaufmann, 1st edition 2014

CO-PO Mapping

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS8001I

Course Title	Soft Computing			
Category	Major ENGG			
LTP & Credits	L	T	P	Credits
	3	1	0	4
Total Contact Hours	36			
Pre-requisites				

Learning Objective:

The main objective of the course is to expose the students to soft computing, various types of soft computing techniques, and applications of soft computing. .Upon completion of this course, the student should be able to get an idea on: Artificial Intelligence, Various types of production systems, characteristics of production systems. Neural Networks, architecture, functions and various algorithms involved. Fuzzy Logic, Various fuzzy systems and their functions.Genetic algorithms, its applications and advances.

Course Outcome:

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

CO1: To learn about soft computing techniques and their applications
CO2: To analyze various neural network architectures
CO3: To understand perceptron and counter propagation networks
CO4: To Define the fuzzy systems
CO5: To analyze the genetic algorithms and their applications

Course Content:

Module 1: **[5L]**

Introduction: Introduction to soft computing; introduction to fuzzy sets and fuzzy logic systems; introduction to biological and artificial neural network; introduction to Genetic Algorithm

Module 2: **[10L]**

Fuzzy sets and Fuzzy logic systems: Classical Sets and Fuzzy Sets and Fuzzy relations : Operations on Classical sets, properties of classical sets, Fuzzy set operations, properties of fuzzy sets, cardinality, operations, and properties of fuzzy relations. Membership functions : Features of membership functions, standard forms and boundaries, different fuzzification methods. Fuzzy to Crisp conversions: Lambda Cuts for fuzzy sets, fuzzy Relations, Defuzzification methods. Classical Logic and Fuzzy Logic: Classical predicate logic, Fuzzy Logic, Approximate reasoning and Fuzzy Implication Fuzzy Rule based Systems: Linguistic Hedges, Fuzzy Rule based system – Aggregation of fuzzy Rules, Fuzzy Inference System- Mamdani Fuzzy Models – Sugeno Fuzzy Models. Applications of Fuzzy Logic: How Fuzzy Logic is applied in Home Appliances, General Fuzzy Logic controllers, Basic Medical Diagnostic systems and Weather forecasting

Module 3: **[10L]**

Neural Network Introduction to Neural Networks: Advent of Modern Neuroscience, Classical AI and Neural Networks, Biological Neurons and Artificial neural network; model of artificial neuron. Learning Methods: Hebbian, competitive, Boltzmann etc., Neural Network models: Perceptron, Adaline and Madeline networks; single layer network; Back propagation and multi-layer networks. Competitive learning networks: Kohonen self organizing networks, Hebbian learning; Hopfield Networks. Neuro-Fuzzy modeling: Applications of Neural Networks: Pattern Recognition and classification

Module 4: **[7L]**

Genetic Algorithms: Simple GA, crossover and mutation, Multi-objective Genetic Algorithm (MOGA). Applications of Genetic Algorithm: genetic algorithms in search and optimization, GA based clustering Algorithm, Image processing and pattern Recognition

Module 5: **[4L]**

Other Soft Computing techniques: Simulated Annealing, Tabu search, Ant colony optimization (ACO), Particle Swarm Optimization (PSO).

Text/Reference Books:

1. Fuzzy logic with engineering applications, Timothy J. Ross, John Wiley and Sons.
2. S. Rajasekaran and G.A.V.Pai, "Neural Networks, Fuzzy Logic and Genetic Algorithms", PHI
3. Principles of Soft Computing, S N Sivanandam, S. Sumathi, John Wiley & Sons
4. Genetic Algorithms in search, Optimization & Machine Learning by David E. Goldberg
5. Neuro-Fuzzy and Soft computing, Jang, Sun, Mizutani, PHI
6. Neural Networks: A Classroom Approach,1/e by Kumar Satish, TMH,

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS8002G			
Course Title	Quantum Computing			
Category	Major ENGG			
LTP & Credits	L	T	P	Credits
	3	1	0	4
Total Contact Hours	36			
Pre-requisites				

Learning Objective:

A basic introduction to quantum mechanics, linear algebra and familiarity with the Dirac notation is provided first to get one's quantum moorings right . This is then followed byan introductory treatment of quantum computation and quantum information covering aspects of quantum entanglement, quantum algorithms, quantum channels. Rudimentary quantum computing is introduced using the IBM quantum computer and associated simulators

Course Outcome:

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

CO1: Understand the fundamental principles of quantum mechanics and linear algebra as applied to quantum computing.
CO2: Explain concepts of quantum entanglement, quantum algorithms, and quantum channels.
CO3: Apply quantum computing principles to solve simple problems using quantum gates and circuits.
CO4: Demonstrate knowledge of quantum cryptography and its applications in secure communication.
CO5: Program a quantum computer using IBMQ and simulators to implement basic quantum measurements and state analysis.

Course Content:

Module 1: **[7L]**
Introduction: Elementary quantum mechanics:, linear algebra for quantum mechanics, Quantum states in Hilbert space, The Bloch sphere, Density operators, generalized measurements, no-cloning theorem.

Module 2: **[7L]**
Quantum correlations: Bell inequalities and entanglement, Schmidt decomposition, super dense coding, teleportation.

Module 3: **[8L]**
Quantum cryptography: quantum key distribution

Module 4: **[7L]**
Quantum gates and algorithms: Universal set of gates, quantum circuits, Solovay-Kitaev theorem, Deutsch-Jozsa algorithm, factoring

Module 5: **[4L]**

Programming a quantum computer: The IBMQ, coding a quantum computer using a simulator to carry out basic quantum measurement and state analysis.

Text/Reference Books:

1. Phillip Kaye, Raymond Laflamme et. al., An introduction to Quantum Computing, Oxford University press, 2007. 2.
2. Chris Bernhardt, Quantum Computing for Everyone, The MIT Press,Cambridge, 2020 3.
3. David McMahon-Quantum Computing Explained-Wiley-Interscience , IEEE Computer Society (2008

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS8002G			
Course Title	Bioinformatics			
Category	Major ENGG			
LTP & Credits	L	T	P	Credits
	3	1	0	4
Total Contact Hours	36			
Pre-requisites				

Learning Objective:

To provide a national bio-information network designed to bridge the inter-disciplinary gaps in biotechnology information. b. To establish link among scientists in organizations involved in R & D and manufacturing activities in biotechnology. c. To build up information resources, prepare database on biotechnology and to develop relevant information handling tools and techniques.

Course Outcome:

CO1: Infer the biological problems using appropriate in silico approaches
CO2: Select the suitable tools or servers to solve the specific biological issue and curate experimental data.
CO3: Perform and analyze database and similarity search sequence alignment.
CO4: Construct and analyze phylogenetic trees
CO5: Use appropriate tools and packages to analyze varied range of biological problems.

Course Content:

Module 1: **[5L]**

Introduction to strings, edit distance strings, string similarity, elementary commands and protocols, Scope of Bioinformatics.

Module 2: **[7L]**

Sequence Databases And Their Use Introduction to databases, database search, algorithms issues in database search, sequence database search, parametric sequence alignments, sub optimal alignments, dynamic programming global and local alignment gaps, multiple alignment, common multiple alignment methods. FASTA and BLAST. Amino acid substitution matrices PAM and BLOSSOM.

Module 3: **[6L]**

Evolutionary Trees And Phylogeny : Ultrasonic trees, parsimony, ultrametric problem, perfect phylogeny, phylogenetic alignment, connection between multiple alignment and tree constructions.
.

Module 4: **[7L]**

Protein Classification And Structure Visualization : Overview of the protein structure, protein structure visualization, visualization tools and databases, protein structure alignment, protein classification approaches, tools for plotting Protein - ligand interaction.

Module 5: **[6L]**

Protein Structure Prediction : Protein identification and characterization, primary structure analysis and prediction, secondary structure analysis and prediction, Ab initio method for protein prediction, protein

function prediction.

Module 6: **[5L]**

Applications Of Bioinformatics: DNA mapping and sequencing, gene predictions, molecular predictions with DNA strings, role of bioinformatics in drug design

Text/Reference Books:

1. David W. Mount. 2005. Bioinformatics: Sequence and Genome analysis, Cold Spring Harbor Laboratory Press.
2. Jones, N.C. and Pevzner, P. A. 2004. An Introduction to Bioinformatics Algorithms. The MIT Press

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YCS8002I			
Course Title	Robotics			
Category	Major ENGG			
LTP & Credits	L	T	P	Credits
	3	0	0	4
Total Contact Hours	36			
Pre-requisites				

Learning Objective:

To study the kinematics, drive systems and programming of robots. To study the basics of robot laws and transmission systems. To familiarize students with the concepts and techniques of robot manipulator, its kinematics. To familiarize students with the various Programming and Machine Vision application in robots. To build confidence among students to evaluate, choose and incorporate robots in engineering systems.

Course Outcome:

CO1: Interpret the features of robots and technology involved in the control
CO2: Apply the basic engineering knowledge and laws for the design of robotics.
CO3: Explain the basic concepts like various configurations, classification and parts of end effectors compare various end effectors and grippers and tools and sensors used in robots
CO4: Explain the concept of kinematics, degeneracy, dexterity and trajectory planning..
CO5: Demonstrate the image processing and image analysis techniques by machine vision system.

Course Content:

Module 1: **[6L]**

Fundamentals Of Robot : Robot – Definition – Robot Anatomy – Co-ordinate systems, Work Envelope, types and classification – specifications – Pitch, yaw, Roll, Joint Notations, Speed of Motion, Pay Load

– Robot Parts and their functions – Need for Robots – Different Applications..

Module 2: **[8L]**

Robot Kinematics Forward kinematics, inverse kinematics and the difference: forward kinematics and inverse Kinematics of Manipulators with two, three degrees of freedom (in 2 dimensional), four degrees of freedom (in 3 dimensional) – derivations and problems. Homogeneous transformation matrices, translation and rotation matrices.

Module 3: **[8L]**

Robot Drive Systems And End Effectors Pneumatic Drives – Hydraulic Drives – Mechanical Drives – Electrical Drives – D.C. Servo Motors, Stepper Motor, A.C. Servo Motors – Salient Features, Applications and Comparison of All These Drives. End Effectors – Grippers – Mechanical Grippers, Pneumatic and Hydraulic Grippers, Magnetic grippers, vacuum grippers, internal grippers and external grippers, selection and design considerations of a gripper

Module 4: **[7L]**

Sensors In Robotics Force sensors, touch and tactile sensors, proximity sensors, non-contact sensors, safety considerations in robotic cell, proximity sensors, fail safe hazard sensor systems, and compliance mechanism. Machine vision system - camera, frame grabber, sensing and digitizing image data – signal conversion, image storage, lighting techniques, image processing and analysis – data reduction, segmentation, feature extraction, object recognition, other algorithms, applications – Inspection, identification, visual serving and navigation.

Module 5: **[7L]**

Programming And Applications Of Robot Teach pendant programming, lead through programming, robot programming languages – VAL programming – Motion Commands, Sensors commands, End-Effector Commands, and simple programs - Role of robots in inspection, assembly, material handling, underwater, space and medical fields

Text/Reference Books:

1. Ganesh.S.Hedge,"A textbook of Industrial Robotics", Lakshmi Publications, 2006.
2. Mikell.P.Groover , "Industrial Robotics – Technology, Programming and applications" McGraw Hill 2ND edition 2012.
3. Fu K.S. Gonalz R.C. and ice C.S.G."Robotics Control, Sensing, Vision and Intelligence", McGraw Hill book co. 2007.

4. YoramKoren, "Robotics for Engineers", McGraw Hill Book, Co., 2002.

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3

Course Code	YED8501			
Course Title	Essence of Indian Knowledge Tradition			
Category	MC			
LTP & Credits	L	T	P	Credits
	3	0	0	0
Total Contact Hours	36			
Pre-requisites				

Learning Objective:

To facilitate the students with the concepts of Indian traditional knowledge and to make them understand the Importance of roots of knowledge system

Course Outcome:

CO1: Understand the concept of Traditional knowledge and its importance.
CO2: Know the need and importance of protecting traditional knowledge
CO3: Know the various enactments related to the protection of traditional knowledge.
CO4: Understand the concepts of Intellectual property to protect the traditional knowledge.
CO5: Understand the traditional knowledge in different sectors.

Course Content:

Module 1: **[8L]**

Introduction to traditional knowledge Define traditional knowledge, nature and characteristics, scope and importance, kinds of traditional knowledge, the physical and social contexts in which traditional knowledge develop, the historical impact of social change on traditional knowledge systems. Indigenous Knowledge (IK), characteristics, traditional knowledge vis-à-vis indigenous knowledge, traditional knowledge Vs western knowledge traditional knowledge vis-à-vis formal knowledge.

Module 2: **[6L]**

Protection of traditional knowledge The need for protecting traditional knowledge Significance of TK Protection, the value of TK in the global economy, Role of Government to harness TK.

Module 3: **[6L]**

Legal framework and Traditional Knowledge (10 Lectures) A: The Scheduled Tribes and Other Traditional Forest Dwellers (Recognition of Forest Rights) Act, 2006, Plant Varieties Protection and Farmers Rights Act, 2001 (PPVFR Act); B: The Biological Diversity Act 2002 and Rules 2004, the protection of traditional knowledge bill, 2016. Geographical indications act 2003.

Module 4: **[8L]**

Traditional knowledge and intellectual property Systems of traditional knowledge protection, Legal concepts for the protection of traditional knowledge, Certain non IPR mechanisms of traditional knowledge protection, Patents and traditional knowledge, Strategies to increase protection of traditional knowledge, global legal FORA for increasing protection of Indian Traditional Knowledge.

Module 5: **[8L]**

Traditional knowledge in different sectors Traditional knowledge and engineering, Traditional medicine system, TK and biotechnology, TK in agriculture, Traditional societies depend on it for their food and healthcare needs, Importance of conservation and sustainable development of environment, Management of biodiversity, Food security of the country and protection of TK.

Text/Reference Books:

1. Traditional Knowledge System in India, by Amit Jha, 2009.
2. Traditional Knowledge System and Technology in India by Basanta Kumar Mohanta and Vipin Kumar Singh, Pratibha Prakashan 2012. .
3. Traditional Knowledge System in India by Amit Jha Atlantic publishers, 2002
4. "Knowledge Traditions and Practices of India" Kapil Kapoor, Michel Danino

CO-PO Mapping:

Course Outcome (CO)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO-1	PSO-2	PSO-3
CO1	3	1	1	2	3	2	0	0	3	0	0	3	0	0	3
CO2	2	2	1	1	1	1	0	0	0	0	0	3	0	0	0
CO3	1	3	1	2	2	1	0	0	1	0	0	3	0	0	3
CO4	2	2	2	1	1	1	0	0	1	0	0	3	0	0	3
CO5	2	1	1	1	2	2	0	0	2	0	0	3	0	0	3