Asansol Engineering College

R25 Curriculum & Syllabus

for B. Tech Under Autonomy

(NEP-2020 Implemented)

Computer Science and Businesss Systems

(CSBS)

(Effective from 2025-26 Admission Batch)

1st Year 1st Semester									
Sl. No.	Broad Catagory	Catagory	Paper Code	Subject	L	T	P	Total	Credit Points
A. Theory									
1	ENGG	Major	CS101	Introduction to Programming and Problem Solving	3	0	0	3	3
2	SCI	Multi	PH101	Engineering Physics	3	0	0	3	3
3	SCI	Multi	M101	Engineering Mathematics-I	3	0	0	3	3
4	HUM	Value Added Course	HU101	Environmental Science	2	0	0	2	2
5	HUM	Value Added Courses	HU102	Indian Knowledge System	1	0	0	1	1
B. Practical									
1	ENGG	Major	CS191	Introduction to Programming and Problem-Solving Lab	0	0	3	3	1.5
2	SCI	Skill Enhancement Course	PH191	Engineering Physics Lab	0	0	3	3	1.5
3	ENGG	Skill Enhancement Course	ME191	Engineering Graphics & Computer Aided Design Lab	0	0	3	3	1.5
4	HUM	Ability Enhancement Course	HU191	Communication & Presentation Skill	0	0	3	3	1.5
C. MANDATORY ACTIVITIES / COURSES									
1	MC	Mandatory Course	MC181	3 Weeks Induction Program*	0	0	0	0	2 Units
Total of Theory, Practical								24	18

***Note:Student Induction Program (SIP)**

The institute is responsible for conducting the Student Induction Program as per the guidelines of the AICTE. Every student must attend the Student Induction Program (SIP) to be conducted by the Institute/Department, the schedule of which is to be circulated on the very first day. The schedule includes the following but is not limited to –

1. Principal’s address.
2. Interaction with parents by the Principal.
3. Physical Activity.

4. Creative Arts and Culture.
5. Mentor-mentee groups meet.
6. Diagnostic test.
7. Lectures on Universal Human Values.
8. Lectures on Social and Emotional Learning.
9. Familiarization with College, Dept./Branch.
10. Literary Activity.
11. Lectures & Workshops by Eminent People and alumni.
12. Visits in the Local Area.
13. Extra-Curricular Activities in College.
14. Feedback and Report on the Program- Presentation of Report by each group.

1st Year 2nd Semester									
Sl. No.	Broad Catagory	Catagory	Paper Code	Subject	L	T	P	Total	Credit Points
A. Theory									
1	ENGG	Major	CS201	Data Structures and Algorithms	3	0	0	3	3
2	ENGG	Minor	CS202	Introduction to Artificial Intelligence	2	0	0	2	2
3	ENGG	Major	CS203	Digital Logic and Computer Organization	3	0	0	3	3
4	SCI	Multi	CH201	Engineering Chemistry	2	0	0	2	2
5	SCI	Multi	M201	Engineering Mathematics –II	3	0	0	3	3
6	HUM	Ability Enhancement Course	HU203	Design Thinking & Innovation	1	0	0	1	1
7	HUM	Value Added Course	HU205	Constitution of India & Professional Ethics	1	0	0	1	1
B. Practical									
1	ENGG	Major	CS291	Data Structures and Algorithms Lab	0	0	3	3	1.5
2	ENGG	Major	CS292	Introduction to Artificial Intelligence Lab	0	0	3	3	1.5
3	ENGG	Major	CS293	Digital Logic and Computer Organization Lab	0	0	3	3	1.5
4	SCI	Skill Enhancement Course	CH291	Engineering Chemistry Lab	0	0	2	2	1
5	ENGG	Skill Enhancement Course	ME292	IDEA LAB Workshop	0	0	3	3	1.5
C. MANDATORY ACTIVITIES / COURSES									
1	MC	Mandatory Course	MC281	NSS /Physical Activities/Meditation / Yoga/Photography or Nature Club	0	0	0	0	2 Units
Total of Theory, Practical								29	22

2nd Year 3rd Semester									
Sl. No.	Broad Catagory	Catagory	Paper Code	Subject	L	T	P	Total	Credit Points
A. Theory									
1	ENGG	Major	CB301	Design Analysis of Algorithms	3	0	0	3	3
2	ENGG	Major	CB302	Object-Oriented Programming Using C++	3	0	0	3	3
3	ENGG	Major	CB303	Database Management Systems	3	0	0	3	3
4	ENGG	Major	CB304	Computer Architecture	3	0	0	3	3
5	HUM	Multi	BS301	Marketing Management	2	0	0	2	2
6	SCI	Multi	M(CB)301	Statistical Methods	3	0	0	3	3
B. Practical									
1	ENGG	Major	CB391	Design Analysis of Algorithms Lab	0	0	3	3	1.5
2	ENGG	Major	CB392	Object-Oriented Programming Using C++ Lab	0	0	3	3	1.5
3	ENGG	Major	CB393	Database Management Systems Lab	0	0	3	3	1.5
4	ENGG	Major	CB394	Introduction to Java Lab	0	0	3	3	1.5
C. MANDATORY ACTIVITIES / COURSES									
1	MC	Mandatory Course	MC381	NSS/ Physical Activities/Meditation & Yoga/Photography or Nature Club	0	0	0	0	2 Units
Total of Theory, Practical								29	23

2nd Year 4th Semester									
Sl. No.	Broad Catagory	Catagory	Paper Code	Subject	L	T	P	Total	Credit Points
A. Theory									
1	ENGG	Major	CB401	Software Engineering and Agile Methodologies	3	0	0	3	3
2	ENGG	Major	CB402	Operating Systems	3	0	0	3	3
3	ENGG	Major	CB403	Computer Networks	3	0	0	3	3
4	ENGG	Major	CB404	Formal Language and Automata Theory	3	0	0	3	3
5	ENGG	Minor	BS401	Principles of Management	3	0	0	3	3
6	SCI	Multi	M(CB)401	Linear Algebra	3	0	0	3	3
B. Practical									
1	ENGG	Major	CB491	Software Engineering and Agile Methodologies Lab	0	0	3	3	1.5
2	ENGG	Major	CB492	Operating Systems Lab	0	0	3	3	1.5
3	ENGG	Major	CB493	Computer Networks Lab	0	0	3	3	1.5
4	HUM	Multi	HU(CB)491	Business Communication and Value Science	1	0	2	3	2
C. MANDATORY ACTIVITIES / COURSES									
1	MC	Mandatory Course	MC481	Course from Skill India-I*	0	0	0	0	2 Units
Total of Theory, Practical								30	24.5

***Note:** Course from Skill India to be done in consultation with the department. Certificate of completion is to be submitted to the department/COE Section by the end of the 4th semester as the mark of completion of MC481.

3rd Year 5th Semester									
Sl. No.	Broad Catagory	Catagory	Paper Code	Subject	L	T	P	Total	Credit Points
A. Theory									
1	ENGG	Major	CB501	Software Design and UML	3	0	0	3	3
2	ENGG	Major	CB502	Machine Learning	3	0	0	3	3
3	ENGG	Major	CB503A	Computer Graphics with Python	3	0	0	3	3
			CB503B	Data Mining and Business Intelligence					
			CB503C	Generative Artificial Intelligence					
			CB503D	Cryptography and Network Security					
3	ENGG	Minor	BA(CB)501A	Financial Computing and Risk Analysis	3	0	0	3	3
			IT(CB)501B	Introduction to Augmented and Virtual Reality					
			IT(CB)501C	E-Commerce and Digital Business Models					
			IT(CB)501D	Soft Computing					
5	HUM	Minor	BS501	Business Accounting and Financial Management	3	0	0	3	3
6	HUM	Minor	BS502	Organizational Behavior	3	0	0	3	3
B. Practical									
1	ENGG	Major	CB591	Software Design and UML Lab	0	0	3	3	1.5
2	ENGG	Major	CB592	Machine Learning Lab	0	0	3	3	1.5
3	ENGG	Major	CB593A	Computer Graphics with Python Lab	3	0	0	3	3
			CB593B	Data Mining and Business Intelligence Lab					
			CB593C	Generative Artificial Intelligence Lab					
			CB593D	Cryptography and Network Security Lab					
4	PRJ	PRJ	CB581	Project-I	0	0	4	4	2
C. MANDATORY ACTIVITIES / COURSES									
1	MC	Mandatory Course	MC581	Course from Skill India-II*	0	0	0	0	2 Units
Total of Theory, Practical								27	24.5

***Note:** Course from Skill India to be done in consultation with the department. Certificate of completion is to be submitted to the department/COE Section by the end of the 5th semester as the mark of completion of MC581.

3rd Year 6th Semester									
Sl. No.	Broad Category	Category	Paper Code	Subject	L	T	P	Total	Credit Points
A. Theory									
1	SCI	Multi	M(CB)601	Computational Statistics	3	0	0	3	3
2	ENGG	Major	CB601	Compiler Design	3	0	0	3	3
3	ENGG	Major	CB602A	Mobile App Development	3	0	0	3	3
			CB602B	Introduction to Internet of Things					
			CB602C	Enterprise Resource Planning (ERP) Systems					
			CB602D	Digital Image Processing					
4	ENGG	Minor	HU(CB)601A	Entrepreneurship and Startup Ecosystem	3	0	0	3	3
			HU(CB)601B	Principles of Marketing in the Digital Era					
			HU(CB)601C	Business Communication and Soft Skills					
			HU(CB)601D	Intellectual Property Rights and Patents					
5	SCI	Multi	M(CB)602	Operation Research	2	0	0	2	2
6	HUM	Minor	BS601	Human Resource Management	3	0	0	3	3
B. Practical									
1	SCI	Multi	M(CB)691	Computational Statistics Lab with R Programming	0	0	3	3	1.5
2	ENGG	Major	CB692A	Mobile App Development Lab	0	0	3	3	1.5
			CB692B	Introduction to Internet of Things (IoT) Lab					
			CB692C	Enterprise Resource Planning (ERP) Systems Lab					
			CB692D	Digitial Image Processing Lab					
3	PRJ	PRJ	CB681	Project-II	0	0	8	8	4
C. Mandatory Activities / Courses									
1	MC	Mandatory Course	MC681	Course from Skill India-III*	0	0	0	0	2 Units
Total of Theory, Practical								31	24

***Note:** Course from Skill India to be done in consultation with the department. Certificate of completion is to be submitted to the department/COE Section by the end of the 5th semester as the mark of completion of MC681.

4th Year 7th Semester									
Sl. No.	**Broad Category**	**Category**	**Paper Code**	**Subject**	**L**	**T**	**P**	**Total**	**Credit Points**
A. Theory									
3	ENGG	Major	CB701A	Big Data Analytics	3	0	0	3	3
			CB701B	Deep Learning and Neural Networks					
			CB701C	Cybersecurity and Ethical Hacking					
			CB701D	Pattern Recognition					
3	ENGG	Major	CB702A	Natural Language Processing	3	0	0	3	3
			CB702B	Introduction to Blockchain					
			CB702C	Quantum Computing Fundamentals					
			CB702D	Introduction to Game Development					
3	ENGG	Major	CB703A	Mobile Computing	3	0	0	3	3
			CB703B	Real-Time Operating Systems					
			CB703C	Introduction to Data Science					
			CB703D	Cloud Computing					
3	ENGG	Minor	HU(CB)701A	Psychology for Engineers	3	0	0	3	3
			HU(CB)701B	Digital Transformation in Business					
			EE(CB)701C	Green Computing					
			IT(CB)701D	Web Technology					
B. Practical									
3	ENGG	Major	CB791A	Big Data Analytics Lab	0	0	3	3	1.5
			CB791B	Deep Learning and Neural Networks Lab					
			CB791C	Cybersecurity and Ethical Hacking Lab					
			CB791D	Pattern Recognition Lab					
3	ENGG	Major	CB792A	Natural Language Processing Lab	0	0	3	3	1.5
			CB792B	Introduction to Blockchain Lab					
			CB792C	Quantum Computing Fundamentals Lab					
			CB792D	Introduction to Game Development Lab					
4	PRJ	PRJ	CB781	Project-III	0	0	12	12	6
Total of Theory, Practical								**27**	**21**

4th Year 8th Semester									
Sl. No.	**Broad Category**	**Category**	**Paper Code**	**Subject**	**L**	**T**	**P**	**Total**	**Credit Points**
A. Theory									
B. Practical									
1	PRJ	PRJ	CB881	Internship or Entrepreneurship or Startup or Proto type Design or Dissertation	0	0	0	0	3
2	ENGG	Major	CB882	Grand Viva	0	0	0	0	2
Total of Theory, Practical								**0**	**5**

1st Year 1st Semester									
Sl. No.	Broad Catagory	Catagory	Paper Code	Subject	L	T	P	Total	Credit Points
A. Theory									
1	ENGG	Major	CS101	Introduction to Programming and Problem Solving	3	0	0	3	3
2	SCI	Multi	PH101	Engineering Physics	3	0	0	3	3
3	SCI	Multi	M101	Engineering Mathematics-I	3	0	0	3	3
4	HUM	Value Added Course	HU101	Environmental Science	2	0	0	2	2
5	HUM	Value Added Courses	HU102	Indian Knowledge System	1	0	0	1	1
B. Practical									
1	ENGG	Major	CS191	Introduction to Programming and Problem-Solving Lab	0	0	3	3	1.5
2	SCI	Skill Enhancement Course	PH191	Engineering Physics Lab	0	0	3	3	1.5
3	ENGG	Skill Enhancement Course	ME191	Engineering Graphics & Computer Aided Design Lab	0	0	3	3	1.5
4	HUM	Ability Enhancement Course	HU191	Communication & Presentation Skill	0	0	3	3	1.5
C. MANDATORY ACTIVITIES / COURSES									
1	MC	Mandatory Course	MC181	3 Weeks Induction Program*	0	0	0	0	2 Units
Total of Theory, Practical								24	18

Course Name: Introduction to Programming and Problem Solving

Course Code: CS101

Contact (Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 3

Prerequisite:

Basic programming overview of class 12.

Course Objective:

The objective of this course is to enable the students to:

- Describe the architecture, memory systems, and evolution of computers.
- Convert between number systems and analyze binary arithmetic including IEEE754 representation.
- Construct algorithms and flowcharts for basic computational problems.
- Implement control structures, arrays, pointers, and functions in C programs.
- Demonstrate structured data types and file I/O using the C programming language.

Course Outcome(s):

After the completion of the course students will be able to

CO1: Describe the architecture, memory hierarchy, and generations of computers, and classify hardware and software components, demonstrating a foundation of engineering knowledge required for understanding computing systems.

CO2: Convert values between number systems and analyze signed and IEEE754 floating-point representations, applying core concepts of mathematics and engineering fundamentals to solve complex engineering problems.

CO3: Construct flowcharts and algorithms for problem solving and develop modular programs in C using appropriate control logic, reflecting skills in design and development of solutions and modern tool usage.

CO4: Implement programs in C using control structures, arrays, pointers, and storage classes, and differentiate between memory management techniques, showcasing proficiency in problem analysis and engineering practice.

CO5: Demonstrate structured data types, file handling, and system-level I/O operations, and evaluate their effectiveness in ensuring data persistence and interfacing with hardware, promoting effective engineering tool usage and lifelong learning.

CO-PO-PSO Mapping

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

Course Content

Module 1:

Basics of Computing & Number Representation (7L)

- History and generations of computers.
- Classification: Digital, Analog, Hybrid, Micro, Mini, Mainframe.
- Computer architecture: Input/Output units, Memory (Primary & Secondary), CPU.
- Number systems: Binary, Octal, Decimal, Hexadecimal.
- Conversions among number systems.
- Signed number representations: 1’s, 2’s complement.
- Floating point representation: IEEE 754 single & double precision.
- ASCII codes.
- Overview of compiler, interpreter, assembler.

Module 2:

Problem Solving & Introduction to C Programming (7 L)

- Algorithm, flowchart, and pseudocode.
- Procedural vs Structured programming.
- C basics: keywords, identifiers, variable naming (Hungarian Notation).
- Data types, constants, declaration, storage size, endianness.
- Operators: Arithmetic, Logical, Relational, Bitwise, Conditional.
- Operator precedence and type conversions.
- Input/Output: scanf(), printf().

Module 3:

Control Structures & Program Design (7 L)

- Control structures: if, if-else, switch, nested conditions.
- Loops: while, for, do-while, break, continue.
- goto and labels (with discussion on structured vs unstructured programming).
- Functions: declaration, definition, prototypes.
- Parameter passing, return types, recursion.
- Storage classes: auto, static, extern, register.
- Preprocessor directives and macros.

Module 4:

Arrays, Pointers and Strings (8 L)

- Arrays: 1D & 2D, array to function passing.
- Pointers: basics, pointer arithmetic, pointer to arrays.
- Strings: character arrays, string library functions, array of strings.
- Dynamic memory allocation: malloc(), calloc(), realloc(), free().

Module 5:

Structured Data Types, File Handling & System Interface (7 L)

- Structures: definition, initialization, array of structures, pointers to structures.
- Unions and enum, typedef, bit fields.
- File I/O in C: fopen(), fclose(), fprintf(), fscanf(), fgetc(), fputc().
- Command line arguments.

Text Books:

1. **”Schaum’s Outline of Programming with C”**, Byron S. Gottfried, McGraw-Hill Education, 1st Edition (1996).
2. **”Let Us C”**,Yashavant Kanetkar, BPB Publications, 17th Edition.
3. **”Computer Fundamentals ”**, P.K. Sinha and Priti Sinha, BPB Publications,6th Edition

Reference Books:

1. **”The C Programming Language”**, Brian W. Kernighan and Dennis M. Ritchiem, Prentice Hall, 2nd Edition.
2. **”Fundamentals of Computers”**, V. Rajaraman and Neeharika Adabala, PHI Learning, 6th Edition.
3. **”Computer Organization and Architecture: Designing for Performance”**, William Stallings, Pearson Education, 10th Edition .
4. **”Mastering C”**, K. R. Venugopal and S. R. Prasad, Tata McGraw-Hill Education, 2nd Edition .
5. **” Programming in ANSI C”**, E. Balagurusamy, McGraw Hill Education 8th Edition.

Course Name: Engineering Physics

Course Code: PH101

Contact (Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 3

Prerequisite:

Knowledge of Physics up to 12^{th} standard.

Course Objective:

The objective of this course is to enable the students to:

- Provide foundational understanding of core physical principles such as optics, quantum mechanics, solid-state physics, and statistical mechanics relevant to engineering disciplines.
- Develop the ability to apply theoretical knowledge of physical sciences in interpreting engineering phenomena and solving problems using scientific reasoning and quantitative analysis.
- Expose students to the working principles of modern devices and technologies like lasers, fiber optics, semiconductors, and nanomaterials used in engineering and industrial applications.
- Encourage scientific curiosity and innovation by connecting physical theories with practical tools and techniques in emerging fields like nanotechnology and quantum systems.
- Understand the role of physics in interdisciplinary domains for the advancement of science, technology, and sustainable development through real-life engineering contexts.

Course Outcome(s):

After the completion of the course students will be able to

CO1: Explain the principles of lasers, fibre optics, and holography and apply them in modern optical and communication systems.

CO2: Identify different crystal structures and compute structural parameters such as Miller indices and packing factors; distinguish between metals, semiconductors, and insulators using band theory.

CO3: Utilize the principles of quantum theory, wave-particle duality, and Schrödinger equation—to interpret fundamental quantum phenomena.

CO4: Illustrate the basic concepts of statistical mechanics and examine their implications on microscopic particle behaviour.

CO5: Describe the properties of nanomaterials and display/storage devices and analyze their applications in modern technology.

CO-PO-PSO Mapping

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

Course Content

Module 1:

Modern Optics (11L)

1.01 **Laser: (6L)** Concepts of various emission and absorption processes, Einstein A and B coefficients and equations, working principle of laser, metastable state, population inversion, condition necessary for active laser action, optical resonator, illustrations of Ruby laser, He-Ne laser, Semiconductor laser, applications of laser, related numerical problems.

1.02 **Fibre Optics: (3L)** Principle and propagation of light in optical fibers (Step index, Graded index, single and multiple modes) - Numerical aperture and Acceptance angle, Basic concept of losses in optical fiber, related numerical problems.

1.03 **Holography:(2L)** Theory of holography (qualitative analysis), viewing of holography, applications

Module 2:

Solid State Physics (5L)

2.01 **Crystal Structure: (3L)**Structure of solids, amorphous and crystalline solids (definition and examples), lattice, basis, unit cell, Fundamental types of lattices –Bravais lattice, simple cubic, fcc and bcc lattices, Miller indices and miller planes, co-ordination number and atomic packing factor, Bragg's equation, applications, numerical problems.

2.01 **Semiconductor: (2L)** Physics of semiconductors, electrons and holes, metal, insulator and semiconductor, intrinsic and extrinsic semiconductor, p-n junction.

Module 3:

Quantum and Statistical Mechanics (14L)

3.01 **Quantum Theory: 5L** Inadequacy of classical physics-concept of quantization of energy, particle concept of electromagnetic wave (example: Black body radiation, Photoelectric and Compton Effect: no derivation required), wave particle duality; phase velocity and group velocity; de Broglie hypothesis; Davisson and Germer experiment, related numerical problems.

3.02 **Quantum Mechanics 1: 4L** Concept of wave function, physical significance of wave function, prob- ability interpretation; normalization of wave functions-Qualitative discussion; uncertainty principle, relevant numerical problems, Introduction of Schrödinger wave equation (only statement).

3.03 **Statistical Mechanics: 5L** Concept of energy levels and energy states, phase space, microstates, macrostates and thermodynamic probability, MB, BE, FD, statistics (Qualitative discussions)-physical significance, conception of bosons, fermions, classical limits of quantum statistics, Fermi distribution at zero & non-zero temperature, Concept of Fermi level-Qualitative discussion.

Module 4:

Physics of Nanomaterials: (4L)

Reduction of dimensionality, properties of nanomaterials, Quantum wells (two dimensional), Quantum wires (one dimensional), Quantum dots (zero dimensional); Quantum size effect and Quantum confinement. Carbon allotropes. Application of nanomaterials (CNT, graphene, electronic, environment, medical)

Module 5:

Storage and display devices (2L)

Different storage and display devices-Magnetic storage materials, Operation and application of CRT, CRO, LED and OLED.

Text Books:

1. **”Concepts of Modern Engineering Physics”**, A. S. Vasudeva, S. Chand Publishers
2. **”Engineering Physics”**,Rakesh Dogra.
3. **”Introduction to Nanoscience and Nanotechnology, An Indian Adaptation”**, Charles P. Poole, Jr., Frank J. Owens.
4. **”Quantum Mechanics”**, S. N. Ghosal.
5. **”Nanotechnology”**, K. K. Chattopadhyay .

Reference Books:

1. **”Optics”**, Ajay Ghatak (TMH).
2. **”Solid state Physics”**, S. O. Pillai.
3. **”Quantum mechanics”**, A.K. Ghatak and S Lokenathan.
4. **”Fundamental of Statistical Mechanics”**, B. B. Laud.
5. **” Perspective & Concept of Modern Physics”**, Arthur Beiser.

Course Name: Engineering Mathematics- I

Course Code: M101

Contact (Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 3

Prerequisite:

The students to whom this course will be offered must have the understanding of (10+2) standard algebraic operations, coordinate geometry, and elementary calculus concepts including limits, continuity, differentiation, and integration.

Course Objective:

The objective of this course is to enable the students to:

- Develop a strong foundation in both fundamental and advanced concepts of linear algebra and calculus essential for engineering applications.
- Build competency in applying integration techniques in multiple dimensions, including line, surface, and volume integrals, to solve problems relevant to engineering and applied sciences.
- Gain proficiency in analyzing multivariable functions using differentiation techniques such as partial and total derivatives, Jacobians, and methods for finding extrema.

Course Outcome(s):

After the completion of the course students will be able to

CO1: Apply linear algebra methods to perform matrix operations, classify matrix structures, solve systems of linear equations, and compute eigenvalues and eigenvectors in engineering contexts.

CO2: Apply differential and integral calculus to evaluate and approximate the behavior of single-variable and multivariable real-valued functions relevant to engineering scenarios.

CO3: Analyze the properties of eigenvalues and eigenvectors to assess matrix diagonalizability and interpret linear transformations using the Cayley-Hamilton theorem in engineering systems.

CO4: Analyze single-variable and multivariable real-valued functions using differential and integral calculus to model and interpret complex behavior in engineering applications.

CO-PO-PSO Mapping

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

Course Content

Module 1:

Liner Algebra (11L)
Echelon form and normal (canonical) form of a matrix; Inverse and rank of a matrix; Consistency and inconsistency of system of linear equations, Solution of system of linear equations; Eigenvalues and eigenvectors; Diagonalization of matrix, Cayley-Hamilton theorem.

Module 2:

Single Variable Calculus (5L)
Rolle's Theorem, Mean value theorems, Taylor's and Maclaurin theorems with remainders; Taylor's series.

Module 3:

Multivariable Calculus (Differentiation) (13L)
Function of several variables; Concept of limit, continuity and differentiability; Partial derivatives, Total derivative and its application; chain rules, Derivatives of implicit functions Euler's theorem on homogeneous function; Jacobian; Maxima and minima of functions of two variables.

Module 4:

Multivariable Calculus (Integration) (7L)
Double Integral, Triple Integral; Change of order in multiple integrals; Line Integral, Surface Integral, Volume Integral. Change of variables in multiple integrals.

Text Books:

1. **"Higher Engineering Mathematics"**, Grewal, B.S., Khanna Publishers, 36*th* Edition, 2010.
2. **"Advanced Engineering Mathematics"**, Kreyszig, E., John Wiley & Sons.

Reference Books:

1. **"A text book of Engineering Mathematics-I"**, Guruprasad, S., New age International Publishers.
2. **"Higher Engineering Mathematics"**, Ramana, B.V., Tata McGraw Hill New Delhi, 11*th* Reprint, 2010.
3. **"Engineering Mathematics for first year"**, Veerarajan, T., Tata McGraw-Hill, New Delhi, 2008.
4. **"A text book of Engineering Mathematics"**, Bali, N.P. and Goyal, M., Laxmi Publications, Reprint, 2008.
5. **" Calculus and Analytic geometry"**, 9*th* Edition, Thomas, G.B. and Finney, R.L., Pearson, Reprint, 2002.
6. **"Calculus, Volumes 1 and 2"**, (2nd Edition), Apostol, M., Wiley Eastern, 1980.
7. **"Linear Algebra - A Geometric approach"**, Kumaresan, S., Prentice Hall of India, 2000.
8. **"Linear Algebra: A Modern Introduction"**,2nd Edition, Poole, D., Brooks/Cole, 2005.
9. **"Schaum's Outline of Matrix Operations"**, Bronson, R., 1988.
10. **"Differential and Integral Calculus, Vol. I & Vol. II,"**, Piskunov, N., Mir Publishers, 1969.

Course Name: Environmental Science

Course Code: HU101

Contact (Periods/Week): 2 periods

Total Contact Hours: 24

Credits: 2

Prerequisite:

Knowledge of 10+2 standard.

Course Objective:

The objective of this course is to enable the students to:

- Realize the importance of environment and its resources.
- Apply the fundamental knowledge of science and engineering to assess environmental and health risk.
- Know about environmental laws and regulations to develop guidelines and procedures for health and safety issues.
- Solve scientific problem-solving related to air, water, land and noise pollution.

Course Outcome(s):

After the completion of the course students will be able to

CO1: Able to understand the natural environment and its relationships with human activities.

CO2: The ability to apply the fundamental knowledge of science and engineering to assess environmental and health risk.

CO3:Ability to understand environmental laws and regulations to develop guidelines and procedures for health and safety issues.

CO4: Acquire skills for scientific problem-solving related to air, water, noise & land pollution.

CO-PO-PSO Mapping

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

Course Content

Module 1:

Resources and Ecosystem (6L)

1. **Resources (4L)**
   Types of resources, Human resource, Population Growth models: Exponential Growth, Logistic growth curve with explanation. Maximum Sustainable Yield [Derivation] Alternative sources of Energy [Solar energy, tidal energy, geothermal energy, biomass energy]

2. **Ecosystem (2L)**
   Components of ecosystem, types of ecosystem, Forest ecosystem, Grassland ecosystem, Desert ecosystem, Pond eco system, Food chain, Food web.

Module 2:

Environmental Degradation (10L)

1. **Air Pollution and its impact on Environment (3L)**
)Air Pollutants, primary & secondary pollutants, Criteria pollutants, Smog, Photochemical smog and London smog, Greenhouse effect, Global Warming, Acid rain, Ozone Layer Depletion.

2. **Water Pollution and its impact on Environment (4L)**
   Water Pollutants, Oxygen demanding wastes, heavy metals, BOD [Rate equation], COD, Eutrophication, Hardness, Alkalinity, TDS and Chloride, Heavy metal (As, Hg, Pb) poisoning and toxicity. Numerical on BOD, Hardness.

3. **Land Pollution and its impact on Environment (1L)**
   Solid wastes, types of Solid Waste, Municipal Solid wastes, hazardous wastes, bio-medical wastes, E-wastes,

4. **Noise Pollution and its impact on Environment (2L)**
   Types of noise, Noise frequency, Noise pressure, Measurement of noise level and decibel (dB) Noise intensity, Noise Threshold limit, Effect of noise pollution on human health. Numerical on Measurement of noise level and decibel (dB) and Noise Threshold limit.

Module 3:

Environmental Management (6L)

1. **Environmental Impact Assessment (1L)**
   IEnvironmental Auditing, Environmental laws and Protection Acts of India, carbon footprint, Green building practices. (GRIHA norms).

2. **Pollution Control and Treatment (2L)**
   Air Pollution controlling devices, Catalytic Converter, Electrostatic Precipitator. WasteWater Treat- ment (Surface water treatment &Activated sludge process), Removal of hardness of water (Temporary &Permanent -Permutitprocess).

3. **Waste Management (3L)**
   Solid waste management, Open dumping, Land filling, incineration, composting & Vermicomposting, E-waste management, and Biomedical Waste management.

Module 4:

Disaster Management (2L)

1. **Study of some important disasters (1L)**
   Natural and Man-made disasters, earthquakes, floods drought, landslide, cyclones, volcanic eruptions, tsunami, oil spills, forest fires.

2. **Disaster Management Techniques (1L)**
   Basic principles of disaster management, Disaster Management cycle, Disaster management policy, Awareness generation program.

Text Books:

1. **”Basic Environmental Engineering and Elementary Biology ”**, Gourkrishna Dasmohapatra, Vikas Publishing.
2. **”Basic Environmental Engineering and Elementary Biology”**,Dr. Monindra Nath Patra & Rahul Kumar Singha, Aryan Publishing House.
3. **”Textbook of Environmental Studies for Undergraduate Courses”**, Erach Barucha for UGC, Universities Press .

Reference Books:

1. **” A Text Book of Environmental Studies”**, Dr. D.K. Asthana & Dr. Meera Asthana, S.Chand Publications.
2. **”Environmental Science (As per NEP 2020)”**, Subrat Roy, Khanna Publisher.

Course Name: Indian Knowledge System

Course Code: HU102

Contact (Periods/Week): 1 periods

Total Contact Hours: 12

Credits: 1

Prerequisite:

A basic knowledge (10+2 level) of Indian history, civilization and culture.

Course Objective:

The objective of this course is to enable the students to:

- Understand the extent and aspects of ancient Indian cultural, philosophical and scientific heritage.
- Explore the philosophical roots of Indian knowledge, the scientific temper and quest for advanced understanding of the universe and deeper knowledge of the self.
- Identify and describe the Indian scientific and technological tools, techniques and discoveries and assess their significance and continuing relevance.
- Develop a liberality and open-mindedness of outlook to foster lifelong learning.
- Acquire the skills to apply traditional knowledge in their everyday lives.

Course Outcome(s):

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

CO1: Define, identify, describe and classify the philosophical, literary and socio-religious heritage of ancient India and the core concepts of the Vedic corpus and way of life.

CO2: Discover, enumerate, compare, contrast and categorize the importance of pioneering developments in science and mathematics and evaluate their continuing relevance.

CO3: Analyze, appraise, correlate and describe the ancient Indian heritage in science and technology and examine technological correlations with present-day technological applications.

CO4: Discover, assess and describe traditional knowledge in health care, architecture, agriculture and other sectors and to explore the history of traditional Indian art forms.

CO-PO-PSO Mapping

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

Course Content

Module 1: An Overview of Indian Knowledge System (3L)

- Importance of Ancient Knowledge
- Definition of IKS
- Classification framework of IKS
- Unique aspects of IKS
- The Vedic corpus: Vedas and Vedangas
- Distinctive features of Vedic life
- Indian philosophical systems: Different schools of philosophy (Orthodox and Unorthodox)

Module 2: Salient Features of the Indian Numeral System (3L)

- Developments in Indian Mathematics in ancient India
- Importance of decimal representation
- The discovery of zero and its importance
- Unique approaches to represent numbers
- Contribution of ancient Indian mathematicians
- Highlights of Indian Astronomy: Historical development and key contributions

Module 3: Indian Science and Technology Heritage (3L)

- Metals and metalworking
- Mining and ore extraction
- Structural engineering and architecture in ancient India: planning, materials, construction and approaches
- Dyes and painting
- Shipbuilding

Module 4: Traditional Knowledge in Different Sectors (3L)

- Traditional knowledge and engineering
- Traditional agricultural practices: resources, methods, technical aids
- Traditional medicine and surgery
- History of traditional art forms and culture

Text Books:

1. **”Traditional Knowledge System in India”**, A. L. Basham, New Delhi: Picador, 2019
2. **”Aspects of Science and Technology in Ancient India.”**,Arun Kumar Jha and Seema Sahay ed, Oxford and New Delhi: Taylor and Francis, 2023.
3. **”Indian Knowledge Systems. Vols. 1 and 2. ”**, Kapil Kapoor and Awadhesh Kumar Singh, New Delhi: D. K. Printworld, 2005.

4. **” History of Astronomy in India”**, S. N. Sen and K. S. Shukla, New Delhi: Indian National Science Academy, 2nd edition, 2000.
5. **” Indian Knowledge System.”**, Arpit Srivastava, AKS University, 2024.

Course Name: Introduction to Programming and Problem Solving Lab

Course Code: CS191

Contact (Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 1.5

Prerequisite:

Knowledge of Physics up to 12^{th} standard.

Course Objective:

By the end of this course, students will be able to:

- Understand the fundamentals of programming logic through algorithmic thinking.
- Implement and debug C programs using various control structures.
- Apply memory management concepts using pointers and arrays.
- Develop structured programs involving functions and recursion.
- Demonstrate file operations and manipulate data using structures and pointers.

Course Outcome(s):

After successful completion of the course, students will be able to:

CO1: Identify fundamental programming constructs such as data types, operators, control structures, and apply them to solve basic computational problems.

CO2: Design modular programs using functions, arrays, and structures, and develop reusable solutions to solve real-world problems.

CO3: Demonstrate the use of pointers and dynamic memory management to analyze memory-efficient solutions for complex problems.

CO4: Construct file-based applications that enable persistent data storage and illustrate communication of results through formatted outputs.

CO5: Integrate multiple programming concepts to create a functional mini-project, demonstrating teamwork, project management skills, and adaptability to emerging challenges.

CO-PO-PSO Mapping

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

Course Content

Lab No.	Title / Experiment	Learning Focus
1	Introduction to C, Basic Input/Output, Data Types, and Operators	I/O operations, operator precedence, expressions
2	Problems on Conditionals: if, if-else, nested if, switch-case	Decision-making constructs
3	Looping Constructs: for, while, do-while	Iterative problem solving
4	Nested Loops: Pattern Printing, Series Problems	Logical structuring using loops
5	Functions: call by value, return types, recursion	Modular programming and recursion
6	Arrays: 1D and 2D array manipulation, search/sort problems	Data storage and iteration
7	Strings: string manipulation functions, array of strings	Character arrays and string operations
8	Pointers: pointer arithmetic, pointers with arrays and functions	Memory-level data access
9	Dynamic Memory Allocation using malloc(), calloc(), free()	Runtime memory management
10	Structures and Unions: defining, accessing, array of structures, pointer to structure	Composite data types and access
11	File I/O: fopen(), fprintf(), fscanf(), fgetc(), fputc()	Persistent data storage
12	Mini Project: Combining structures, functions, and file I/O for a real-world scenario	Integration and application of concepts

Text Books:

1. Practical Physics by Chatterjee & Rakshit (Book & Allied Publisher).
2. Practical Physics by K.G. Mazumder (New Central Publishing).
3. Practical Physics by R. K. Kar (Book & Allied Publisher).

Course Name: Engineering Physics Lab

Course Code: PH191

Contact (Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 1.5

Prerequisite:

Knowledge of Physics up to 12^{th} standard.

Course Objective:

- Become familiar with scientific instruments and measurement techniques used to determine various physical parameters of materials and systems.
- Reinforce theoretical concepts learned in classroom physics by performing related practical experiments and observing real-time outcomes.
- Develop a systematic and analytical approach to collecting, organizing, and interpreting experimental data for error analysis and validation of physical laws.
- Engage in the experimental validation of physical laws through laboratory activities involving classical mechanics, optics, electronics, and quantum phenomena.
- Encourage innovation and problem-solving abilities through hands-on investigation of advanced and application-oriented physics experiments, including specially designed extension activities.

Course Outcome(s):

After the completion of the course students will be able to

CO1: Determine mechanical properties such as Young’s modulus and rigidity modulus through hands-on experiments and analyze material behaviour under applied forces

CO2: Perform optical experiments including Newton’s Rings, laser diffraction, and optical fiber characterization, and interpret the results based on wave optics principles.

CO3: Investigate quantum effects such as the photoelectric effect and atomic transitions, and relate experimental outcomes to basic quantum principles.

CO4: Study the performance of semiconductor and electronic devices like solar cells, LEDs, and LCR circuits, and investigate their operational characteristics.

CO5: Conduct experiments such as Hall Effect, e/m determination, prism dispersion, or optical rotation to demonstrate the application of advanced physical principles in practical scenarios.

CO-PO-PSO Mapping

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

Course Content

Module 1:

General idea about Measurements and Errors (One Mandatory)
Error estimation using Slide callipers/ Screw-gauge/travelling microscope for one experiment.

Module 2:

Experiments on Classical Physics (Any 4 to be performed from the following experiments)

1 Study of Torsional oscillation of Torsional pendulum & determination of time using various load of the oscillator.

2 Determination of Young’s moduli of different materials.

3 Determination of Rigidity moduli of different materials.

4 Determination of wavelength of light by Newton’s ring method.

5 Determination of wavelength of light by Laser diffraction method.

6 Optical Fibre-numerical aperture, power loss.

Module 3:

Experiments on Quantum Physics (Any 2 to be performed from the following experiments)

7 Determination of Planck’s constant using photoelectric cell.

8 Verification of Bohr’s atomic orbital theory through Frank-Hertz experiment.

9 Determination of Stefan’s Constant.

10.a. Study of characteristics of solar cell (illumination, areal, spectral).

10.b. Study of characteristics of solar cell (I-V characteristics, Power-load characteristics, Power-wavelength characteristics)

Module 4:

Perform at least one of the following experiments

11 Determination of Q factor using LCR Circuit.

12 Study of I-V characteristics of a LED/LDR.

13 Determination of band gap of a semiconductor.

**In addition, it is recommended that each student should carry out at least one experiment beyond the syllabus/one experiment as Innovative experiment.

Module 5:

Probable experiments beyond the syllabus

1 Determination of the specific charge of the electron (e/m) from the path of an electron beam by Thomson method.

2 Determination of Hall co-efficient of a semiconductor and measurement of Magnetoresistance of a given semiconductor, Study of dispersive power of material of a prism.

3 Determination of thermal conductivity of a bad/good conductor using Lees-Charlton / Searle apparatus.

4 Determination of the angle of optical rotation of a polar solution using polarimeter.

5 Any other experiment related to the theory.

Text Books:

1. Practical Physics by Chatterjee & Rakshit (Book & Allied Publisher).
2. Practical Physics by K.G. Mazumder (New Central Publishing).
3. Practical Physics by R. K. Kar (Book & Allied Publisher).

Course Name: Engineering Graphics & Computer Aided Design Lab

Course Code: ME191

Contact (Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 1.5

Prerequisite:

Basic knowledge of geometry.

Course Objective:

- To learn all the skills associated with the tools and inventory associated with the IDEA Lab.
- Learn useful mechanical and electronic fabrication processes.
- Learn necessary skills to build useful and standalone system/ project with enclosures.
- Learn necessary skills to create print and electronic documentation for the system/project.

section*Course Outcome(s): After the completion of the course students will be able to

CO1: Use common drafting tools with the knowledge of drafting standards.

CO2: Understand the concepts of engineering scales, projections, sections.

CO3: Apply computer aided drafting techniques to represent line, surface or solid models in different Engineering viewpoints.

CO4: Produce part models; carry out assembly operation and represent a design project work.

CO-PO-PSO Mapping

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

Course Content

Basic Engineering Graphics (3P)

- 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

Module 1: Introduction to Engineering Drawing (6P)

- Principles of Engineering Graphics and their significance
- Usage of Drawing Instruments, Lettering
- Conic Sections including Rectangular Hyperbola (General method only)
- Cycloid, Epicycloid and Involute
- Scales – Plain, Diagonal and Vernier Scales

Module 2: Orthographic & Isometric Projections (6P)

- Principles of Orthographic Projections – Conventions
- Projections of Points and Lines inclined to both Planes
- Projections of Planes on Inclined Planes – Auxiliary Planes
- Projection of Solids inclined to both the Planes – Auxiliary Views
- Isometric Scale and Views of Lines, Planes, Simple and Compound Solids
- Conversion of Isometric Views to Orthographic Views and Vice-versa

Module 3: Sections and Sectional Views of Right Angular Solids (6P)

- Drawing Sectional Views of Solids – Prism, Cylinder, Pyramid, Cone
- Project the True Shape of the Sectioned Surface
- Auxiliary Views
- Development of Surfaces of Right Regular Solids – Prism, Pyramid, Cylinder, Cone
- Sectional Orthographic Views of Objects from Industry and Dwellings (Foundation to Slab only)

Computer Graphics (3P)

- Engineering Graphics Software
- Spatial Transformations
- Orthographic Projections
- Model Viewing
- Coordinate Systems
- Multi-view Projection
- Exploded Assembly
- Animation, Surface & Solid Modeling
- Spatial Manipulation

Module 4: Overview of Computer Graphics (3P)

- Demonstration of CAD Software:
  - Menu System, Toolbars (Standard, Properties, Draw, Modify, Dimension)
  - Drawing Area (Background, Crosshairs, Coordinate System)
  - Dialog Boxes and Windows
  - Shortcut Menus (Button Bars), Zooming Methods
  - Select and Erase Objects

Module 5: CAD Drawing, Customization, Annotations, Layering (6P)

- Drawing Page Setup including Scale Settings, ISO and ANSI Standards
- Drawing Methods – Straight Lines, Circles
- Applying Dimensions and Annotations
- Layers – Creation and Management
- Line Editing (Extend/Lengthen)
- Drawing Sectional Views of Solids
- Annotations, CAD Modeling of Parts and Assemblies with Animation
- Parametric and Non-parametric Solid, Surface and Wireframe Modeling
- Part Editing and Printing Documents

Module 6: Demonstration of a Simple Team Design Project (3P)

- Illustrating Geometry and Topology of Engineered Components
- Creation of Engineering Models
- Presentation in Standard 2D Blueprint Form and 3D Wireframe/Shaded Solids
- Use of Solid-Modeling Software for Component and Assembly-level Models

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.

Course Name: Communication and Presentation Skill

Course Code: HU191

Contact (Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 1.5

Prerequisite:

Basic knowledge of LSRW skills

Course Objective:

The objectives of the course are to make the students able to-

- Acquire interpersonal communication skills of listening comprehension and speaking in academic and professional situations.
- Understand English pronunciation basics and remedy errors.
- Operate with ease in reading and writing interface in global professional contexts.
- Deliver professional presentations before a global audience.
- Develop confidence as a competent communicator.

Course Outcome(s):

After the completion of the course students will be able to

CO1: Recognize, identify and express advanced skills of Technical Communication in English and Soft Skills through Language Laboratory.

CO2: Understand, categorize, differentiate and infer listening, speaking, reading and writing skills in societal and professional life.

CO3: Analyze, compare and adapt the skills necessary to be a competent interpersonal communicator in academic and global business environments.

CO4: Deconstruct, appraise and critique professional writing documents, models and templates.

CO5: Adapt, negotiate, facilitate and collaborate with communicative competence in presentations and work-specific conclaves and interactions in the professional context.

CO-PO-PSO Mapping

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

Course Content

Module 1: Introduction – Theories of Communication and Soft Skills

- Communication and the Cyclic Process of Communication (Theory, benefits and application)
- Introduction to Workplace Communication (Principles and Practice)
- Non-Verbal Communication and its Application
- **Soft Skills Introduction**:
  - What is Soft Skills? Significance of Soft Skills
  - Soft Skills vs. Hard Skills
  - Components of Soft Skills
  - Identifying and Exhibiting Soft Skills (Through classroom activity)

Module 2: Active Listening

- What is Active Listening?
- Listening Sub-skills – Predicting, Clarifying, Inferencing, Evaluating, Note-taking
- Differences between Listening and Hearing, Critical Listening, Barriers to Active Listening, Improving Listening
- Listening in Business Telephony and Practice
- **Practical**: Role plays, Case studies

Module 3: Speaking Skills

- **Effective Public Speaking**:
  - Selecting the topic, understanding the audience
  - Organizing the main ideas, language and style, delivering the speech
  - Voice clarity
  - **Practical:** Extempore
  - **Self-learning Topics:** Preparation, Attire, Posture and Delivery techniques
- Pronunciation Guide – Basics of Sound Scripting, Stress and Intonation
- Fluency-focused Activities – JAM, Conversational Role Plays, Speaking using Picture/Audio-Visual Inputs
- Group Discussion: Principles, Do’s and Don’ts, and Practice

Module 4: Writing and Reading Comprehension

- Reading and Writing a Book Review (classroom activity)
- Writing a Film Review after watching a short film (classroom activity)
- Reading Strategies: Active Reading, Note-taking, Summarizing, Using Visual Aids like Diagrams and Graphs
- Solving Company-Specific Verbal Aptitude Papers (Synonyms, Antonyms, Error Correction, RC Passages)

Module 5: Presentation Skills

- Kinds of Presentation, Presentation Techniques
- Planning and Structuring the Presentation: Preparation, Research, Evidence
- Delivering the Presentation, Handling Questions, Time Management, Visual Aids
- Self Introduction, Creation of Video R´esum´e
- Need for Expertise in Oral Presentation
- **Assignment:** Oral Presentation
- Rules of Making a Micro Presentation (PowerPoint)
- **Assignment:** Micro Presentation

Text Books

1. Pushp Lata and Sanjay Kumar. *A Handbook of Group Discussions and Job Interviews*. New Delhi: PHI, 2009.
2. Jo Billingham. *Giving Presentations*. New Delhi: Oxford University Press, 2003.
3. B. Jean Naterop and Rod Revell. *Telephoning in English*. 3rd ed. Cambridge: Cambridge University Press, 2004.
4. Jeyaraj John Sekar. *English Pronunciation Skills: Theory and Praxis*. New Delhi: Authorspress, 2025.
5. Career Launcher. *IELTS Reading: A Step-by-Step Guide*. G. K. Publications, 2028.

Reference Books

1. Ann Baker. *Ship or Sheep? An Intermediate Pronunciation Course*. Cambridge University Press, 2006.
2. Barry Cusack and Sam McCarter. *Improve Your IELTS: Listening and Speaking Skills*. Macmillan, 2007.
3. Eric H. Glendinning and Beverly Holmström. *Study Reading*. Cambridge University Press, 2004.
4. Malcolm Goodale. *Professional Presentations*. Cambridge University Press, 2005.
5. Mark Hancock. *English Pronunciation in Use*. Cambridge University Press, 2003.
6. Tony Lynch. *Study Listening*. Cambridge University Press, 2004.
7. J. D. O'Connor. *Better English Pronunciation*. Cambridge University Press, 2005.
8. Peter Roach. *English Phonetics and Phonology: A Practical Course*. Cambridge University Press, 2000.

1st Year 2nd Semester									
Sl. No.	Broad Catagory	Catagory	Paper Code	Subject	L	T	P	Total	Credit Points
A. Theory									
1	ENGG	Major	CS201	Data Structures and Algorithms	3	0	0	3	3
2	ENGG	Minor	CS202	Introduction to Artificial Intelligence	2	0	0	2	2
3	ENGG	Major	CS203	Digital Logic and Computer Organization	3	0	0	3	3
4	SCI	Multi	CH201	Engineering Chemistry	2	0	0	2	2
5	SCI	Multi	M201	Engineering Mathematics –II	3	0	0	3	3
6	HUM	Ability Enhancement Course	HU203	Design Thinking & Innovation	1	0	0	1	1
7	HUM	Value Added Course	HU205	Constitution of India & Professional Ethics	1	0	0	1	1
B. Practical									
1	ENGG	Major	CS291	Data Structures and Algorithms Lab	0	0	3	3	1.5
2	ENGG	Major	CS292	Introduction to Artificial Intelligence Lab	0	0	3	3	1.5
3	ENGG	Major	CS293	Digital Logic and Computer Organization Lab	0	0	3	3	1.5
4	SCI	Skill Enhancement Course	CH291	Engineering Chemistry Lab	0	0	2	2	1
5	ENGG	Skill Enhancement Course	ME292	IDEA LAB Workshop	0	0	3	3	1.5
C. MANDATORY ACTIVITIES / COURSES									
1	MC	Mandatory Course	MC281	NSS /Physical Activities/Meditation / Yoga/Photography or Nature Club	0	0	0	0	2 Units
Total of Theory, Practical								29	22

Course Name: Data Structures and Algorithms

Course Code: CS201

Contact (Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 3

Prerequisite:

1. Familiarity with the fundamentals of C or other programming language.
2. A solid background in mathematics, including probability, set theory.

Course Objective:

By the end of this course, students will be able to:

- Gain a strong foundation in data abstraction, data types, and data structures, and understand the importance of structured data organization in solving engineering problems.
- Formulate and analyze algorithms, perform asymptotic analysis using the notation Big O, ϑ (Theta) and Ω (Omega), and understand the trade-offs between the complexities of time and space.
- Design and implement linear and non-linear data structures such as arrays, linked lists, stacks, queues, trees, heaps, and graphs, and apply them effectively in computational problem-solving.
- Evaluate and compare various searching, sorting, and hashing algorithms based on their performance, and choose appropriate methods for optimized data handling.
- Appreciate the role of data structures in real-world applications, foster a mindset of lifelong learning, and develop the adaptability to utilize modern programming tools and emerging technologies.

Course Outcome(s):

After the completion of the course students will be able to

CO1: Apply fundamental knowledge of data types, abstract data types, and data structures to analyze real-world computational problems and their memory/time constraints.

CO2: Design and implement linear data structures (arrays, linked lists, stacks, queues) using appropriate programming constructs to solve well-defined problems efficiently.

CO3: Develop recursive algorithms and simulate stack-based computations such as expression conversion and evaluation using appropriate engineering tools.

CO4: CO4 Construct and evaluate non-linear data structures (Binary Tree, BST, AVL Tree, heaps, graphs) and associated operations (search, insertion, deletion, traversal) to address complex engineering problems.

CO5: Compare and optimize sorting, searching, and hashing algorithms based on performance analysis and recognize their suitability in dynamic problem contexts to support life-long learning.

CO-PO-PSO Mapping

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

Course Content

Module 1: Introduction (4L)

- Concepts of Data and Information
- Abstract Data Type (ADT), Data Structure and Data Type
- Classification of Data Structures:
  - Primitive and Non-Primitive Data Structures
  - Linear and Non-Linear Data Structures
- Need for Data Structures
- Concept of Algorithms and Programs
- Methods of Representing Algorithms
- Algorithm Analysis: Time and Space Complexity
- Asymptotic Notations:
  - Big Oh O
  - Small Oh o
  - Big Omega Ω
  - Small Omega ω
  - Theta Θ

  (Definition and Significance)

Module 2: Non-Restricted Linear Data Structure (9L)

- **List or Linear List**:
  - Definition, Example, List as ADT
  - Sequential and Linked Representation
- **Array**:
  - Sequential Representation, Linearization of Multidimensional Array
  - Applications: Polynomial Representation, Sparse Matrix Representation
- **Linked List**:
  - Introduction, Implementation of:

    - Singly Linked List
    - Doubly Linked List
    - Circular Linked List
    - Circular Doubly Linked List
  - Application: Polynomial Representation

Module 3: Restricted Linear Data Structure (6L)

- **Stack**:
  - Definition and Implementation using Array and Linked List
  - Applications: Infix to Postfix Conversion, Postfix Evaluation
- **Recursion**:
  - Principles, Use of Stack, Tail Recursion
  - Tower of Hanoi Problem
- **Queue**:
  - Definition, Array Implementation (Physical, Linear, Circular Models)
  - Linked List Implementation
  - Dequeue – Definition and Types

Module 4: Nonlinear Data Structures (9L)

- **Trees and Binary Tree**:
  - Basic Terminologies, Differences between Tree and Binary Tree
  - Representations (Array and Linked List)
  - Traversals: Pre-, In-, Post-order
  - Threaded Binary Tree: Definition, Insertion, Deletion
  - Binary Search Tree (BST): Definition, Insertion, Deletion, Searching
  - AVL Tree (Height Balanced Tree): Definition, Insertion, Deletion (Examples Only)
- **m-Way Search Tree**:
  - B Tree: Definition, Insertion, Deletion (Examples Only)
  - B+ Tree: Definition, Insertion, Deletion (Examples Only)
- **Heap**:
  - Definition (Min and Max Heap), Creation, Insertion, Deletion
  - Application: Priority Queue, Sorting
- **Graphs**:
  - Definition, Representations (Adjacency Matrix, Incidence Matrix, Adjacency List)
  - Traversals: Depth-First Search (DFS), Breadth-First Search (BFS)
  - Edge Types in DFS/BFS: Tree, Back, Cross, Forward Edges

Module 5: Sorting and Searching (8L)

- **Sorting**:
  - Definition, Need for Sorting
  - Types: Internal, External, Stable, In-place, Comparison-based
  - Factors Affecting Sorting
  - Algorithms (with Time Complexity):
    - Bubble Sort
    - Insertion Sort
    - Selection Sort
    - Quick Sort
    - Merge Sort
    - Radix Sort
- **Searching**:
  - Factors Affecting Searching
  - Sequential Search with and without Sentinel (Time Analysis)
  - Binary Search and Interpolation Search (Time Analysis)
- **Hashing**:
  - Introduction, Purpose of Hashing
  - Hash Functions: Division, Folding, Mid-Square
  - Collision Resolution Techniques

Text Books

1. Samiran Chattopadhyay, Debabrata Ghosh Dastidar, Matangini Chattopadhyay, *Data Structures Through 'C' Language*, BPB Publications, Edition: 2001.
2. Ellis Horowitz, Sartaj Sahni, Susan Anderson-Freed, *Fundamentals of Data Structures in C*, 2nd Edition, Universities Press.

Reference Books

1. Thomas A. Standish, *Data Structures, Algorithms, and Software Principles in C*, 1st Edition, Pearson.
2. Seymour Lipschutz, *Data Structures*, Special Indian Edition, Tata McGraw Hill Education.
3. Robert L. Kruse, Bruce P. Leung, *Data Structures and Program Design in C*, 2nd Edition, Pearson.
4. Aaron M. Tenenbaum, *Data Structures in C*, 1st Edition, Pearson.

Course Name: Introduction to Artificial Intelligence

Course Code: CS202

Contact (Periods/Week): 2 periods

Total Contact Hours: 30

Credits: 2

Prerequisite:

Basic Computer Knowledge.

Course Objective:

The objective of the course is to make the students able to –

- Comprehend the fundamental concepts of Knowledge Representation and Inferencing in Artificial Intelligence and its utilitarian importance in current technological context.
- FFormulate a problem as State-Space Exploration Framework or an Inferencing Framework of Artificial Intelligence.
- Use the strategies of AI-Heuristics to find acceptable solutions avoiding brute-force techniques.
- Evaluate and compare various searching, sorting, and hashing algorithms based on their performance, and choose appropriate methods for optimized data handling.
- Design AI-Frameworks for Inferencing based on knowledge base.
- Analyse the effectiveness of AI-Inferencing Model in offering solutions to the respective problem.

Course Outcome(s):

After the completion of the course students will be able to

CO1: Understand and explain the fundamental concepts of Knowledge Representation and Inferencing in Artificial Intelligence and its utilitarian importance in current technological context for further exploration leading towards lifelong learning.

CO2: Identify and formulate an engineering problem primarily to fit a State-Space Exploration Framework or an Inferencing Model/Agent Design Framework within the scope of Artificial Intelligence paradigm.

CO3: Explore relevant literature and apply the concept of Heuristic Techniques of Artificial Intelligence to solve problems.

CO4: Develop Inferencing Models for proposing solutions to the problems of Artificial Intelligence.

CO5: Implement Inferencing Models of Artificial Intelligence through developing feasible algorithms and investigate their effectiveness by analysing their performances in solving the relevant problems.

CO-PO-PSO Mapping

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

Course Content

Module 1: Introduction to Artificial Intelligence (3 Lectures)

- Why AI
- Definition of AI
- Goals of AI
- History and evolution of AI
- Types of AI: Narrow, General, Super
- Human vs Artificial Intelligence
- Applications of AI in various domains
- AI for social good

Module 2: Intelligent Agents and Logic-Based Thinking (8 Lectures)

- Intelligent systems
- Agents and environments
- Decision making using rules and logic
- Symbolic AI concepts
- Propositional Logic: Knowledge Representation and Inference
- Predicate Logic: Knowledge Representation, Inference, and Answer Extraction

Module 3: Overview of AI Branches and Perception (8 Lectures)

- Machine Learning
- Deep Learning
- Natural Language Processing
- Computer Vision
- Expert Systems
- Fuzzy Logic

- Evolutionary Algorithms
- Reinforcement Learning
- Planning and Scheduling
- Human-AI Collaboration

Module 4: Basics of Machine Learning (6 Lectures)

- What is Machine Learning
- AI vs ML
- Types of learning: Supervised, Unsupervised
- Concept of dataset, features, and labels
- ML model and prediction flow
- Common ML applications
- Introduction to decision trees (concept only)
- ML pipeline overview

Module 5: Applications and Ethics of AI (5 Lectures)

- AI in robotics and automation
- AI-enabled smart applications
- Industry 4.0 and intelligent systems
- AI in different sectors: Healthcare, Agriculture, Transport, Education, etc.
- Human-AI teamwork
- Basics of AI ethics: Bias, Fairness, Privacy
- Career opportunities and future scopes in AI

Text Books

1. Saptarsi Goswami, Amit Kumar Das, Amlan Chakrabarti, *AI for Everyone: A Beginner’s Handbook for Artificial Intelligence*, Pearson.
2. Rich, E., Knight, K., Shankar, B., *Artificial Intelligence*, Tata McGraw Hill, 3rd Edition, 2009.
3. Russell, S., Norvig, P., *Artificial Intelligence - A Modern Approach*, Prentice Hall, 3rd Edition, 2015.

Reference Books

1. Reema Thareja, *Artificial Intelligence: Beyond Classical AI*, Pearson.
2. Patterson, *Introduction to Artificial Intelligence and Expert Systems*, Pearson.

Course Name: Digital Logic and Computer Organization

Course Code: CS203

Contact (Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 3

Prerequisite:

Basic knowledge of programming fundamentals.

Course Objective:

By the end of this course, students will be able to:

- To introduce number systems, logic gates, and design of combinational and sequential circuits.
- To develop an understanding of data processing using micro-operations and instruction formats.
- To explain how CPU, memory, and I/O units are organized and interact during instruction execution.
- To describe arithmetic algorithms and control unit designs in processor architecture.
- To build a foundation for advanced topics like microprocessors, computer architecture, and embedded systems.

Course Outcome(s):

After the completion of the course students will be able to

CO1: Explain various number systems and coding schemes, and apply Boolean algebra laws and Karnaugh Maps to simplify logical expressions.

CO2: Design and construct combinational and sequential logic circuits including adders, multiplexers, flip-flops, and counters for implementing digital functions.

CO3: Develop and analyze data path units such as ALU, control units, and register organizations to support instruction execution in CPU architectures.

CO4: CO4 Demonstrate arithmetic operations like Booth’s multiplication and division, and illustrate various addressing modes and instruction formats used in CPUs.

CO5: Compare memory hierarchy systems and I/O techniques, and evaluate their role in enhancing overall processor performance.

CO-PO-PSO Mapping

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

Course Content

Module 1: Number Systems, Boolean Algebra, and Logic Simplification (6 Lectures)

- Binary, BCD, ASCII, EBCDIC, Gray Code and conversions [1L]
- Boolean Algebra – Laws, Theorems [1L]
- Boolean Functions, Minterm and Maxterm, SOP and POS Forms [2L]
- Karnaugh Map (up to 4-variable), Algebraic Simplification [2L]

Module 2: Combinational Circuits (6 Lectures)

- Half and Full Adder/Subtractor, Serial and Parallel Adders, CLA Adder [2L]
- Parity Generator, Encoder, Decoder, Multiplexer, Demultiplexer [2L]
- Comparator, Code Converters [2L]

Module 3: Sequential Circuits and Registers (6 Lectures)

- Flip-Flops: SR, JK, Master-Slave JK, D, T; Characteristic and Excitation Tables [2L]
- Counters: Synchronous/Asynchronous, Ring and Johnson, Mod-N Counters [2L]
- Registers: SISO, SIPO, PIPO, PISO [1L]
- Applications of Counters and Registers [1L]

Module 4: Data Representation and Arithmetic Operations (5 Lectures)

- Integer Arhmetic (Addition, Subtraction), Booth’s Multiplication Algorithm [2L]
- Restoring and Non-Restoring Division [1L]
- Instruction Formats and Addressing Modes [2L]

Module 5: CPU and Control Unit Organization (6 Lectures)

- Register Transfer Language (RTL), Bus Architecture, Micro-operations [1L]
- ALU Design, Status Flags, General Register and Stack Organization [2L]
- Control Unit: Hardwired vs. Microprogrammed Control, Sequencing [2L]
- Basic Instruction Cycle and Execution Pipeline [1L]

Module 6: Memory and I/O Organization (7 Lectures)

- RAM, ROM Types, Memory Hierarchy: Cache, Main, Secondary [1L]
- Cache Mapping: Direct, Associative, Set-Associative; Write Policies [3L]
- Virtual Memory: Paging, Segmentation, FIFO and LRU [1L]
- I/O Transfer Modes: Programmed I/O, Interrupt-Driven I/O, DMA [1L]
- Interrupts: Maskable/Non-Maskable, Daisy Chaining; I/O Processor [1L]

Textbooks

1. *Digital Logic and Computer Design*, M. Morris Mano, Pearson Education, 1st Edition.
2. *Computer Organization and Architecture: Designing for Performance*, William Stallings, Pearson Education, 10th Edition.

Reference Books

1. *Digital Design*, M. Morris Mano, Michael D. Ciletti, Pearson Education, 5th Edition.
2. *Computer Organization and Embedded Systems*, Carl Hamacher, Zvonko Vranesic, Safwat Zaky, McGraw-Hill Education, 6th Edition.
3. *Computer Organization and Design: The Hardware/Software Interface*, David A. Patterson, John L. Hennessy, Morgan Kaufmann Publishers, RISC-V Edition.
4. *Fundamentals of Logic Design*, Charles H. Roth Jr., Larry L. Kinney, Cengage Learning, 7th Edition.
5. *Digital Fundamentals*, Thomas L. Floyd, Pearson Education, 11th Edition.

Course Name: Engineering Chemistry

Course Code: CH201

Contact (Periods/Week): 2 periods

Total Contact Hours: 24

Credits: 2

Prerequisite:

10+2

Course Objective:

- Understand the basic principles of atomic structures and periodic properties of elements,different engineering materials, advanced polymers.
- Apply the knowledge of free energy, energy storage device and semiconductors to design environment friendly and sustainable devices.
- Apply the concept of corrosion and fuel to improve its efficacy and application for industrial purpose.
- Analyze the organic reaction with the structure of organic molecules by applying the knowledge of different spectroscopic techniques.
- Evaluate the electrical, optical, and structural properties of semiconductors to analyze their potential applications in modern electronic and energy devices.

Course Outcome(s):

After the completion of the course students will be able to

CO1: Understand the basic principles of atomic structures and periodic properties of elements, different engineering materials, advanced polymers.

CO2: Apply the knowledge of free energy, energy storage device and semiconductors to design environment friendly and sustainable devices.

CO3: Utilize the concept of corrosion and fuel to improve its efficacy and application for industrial purpose.

CO4: Analyze the organic reaction with the structure of organic molecules by applying the knowledge of different spectroscopic techniques.

CO5: Evaluate the electrical, optical, and structural properties of semiconductors to analyze their potential applications in modern electronic and energy devices.

CO-PO-PSO Mapping

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

Course Content

Module 1 (6 Lectures)

Quantum Properties of Atoms (4L)

- Schrodinger Wave Equation (time independent – basic principles only)
- de Broglie Equation
- Heisenberg Uncertainty Principle
- Quantum Numbers
- Effective nuclear charge, Slater's rule
- Penetration of orbitals, variation of orbital energies in the periodic table
- Atomic and ionic sizes, ionization energies, electron affinity and electronegativity
- Oxidation properties

Chemistry of Materials (2L)

- Semiconductor-Based Memory Materials (Si & Ge): Introduction, Properties, and Roles
- Intensive and Extensive semiconductor properties

Module 2 (7 Lectures)

Chemical Thermodynamics (5L)

- First and Second Law of Thermodynamics
- Tendency for maximum randomness
- Carnot Heat Engine (Derivation)
- Entropy characteristics, mathematical explanation and physical significance
- Entropy change of ideal gas for isothermal reversible process
- Gibbs free energy function, Standard free energy
- Criterion of spontaneity

Electricity Production through Chemical Reactions (2L)

- Electrochemical Cell, cell notation
- Free energy and EMF, Criterion of spontaneity
- Nernst Equation (only expression, no derivation) and applications
- EMF of a cell, single electrode potential, calculation of K_c and K_c from ΔG^0
- Working principle and applications of Lithium-ion batteries

Module 3 (6 Lectures)

Polymers for Engineering Applications (3L)

- Classification of polymers (based on origin, chemical structure, polymeric structure, tacticity, and molecular forces)
- Synthesis and applications of Bakelite, Nylon 6,6, HDPE and LDPE
- Conducting polymers – types, examples, and applications
- Biodegradable polymers – definition, examples, and uses

Industrial Chemistry (3L)

- Types of corrosion, Electrochemical theory of corrosion, rusting of iron
- Comparison of chemical and electrochemical corrosion (mechanism excluded)
- Factors affecting corrosion rate: nature of metal (physical state, purity, position in Galvanic series), environment
- Corrosion control methods: Cathodic protection, Anodic protection, Inorganic coatings
- Classification of fuels (LPG, CNG, BIOGAS), Definitions of Calorific value, Octane number, Cetane number, HCV, LCV

Module 4 (5 Lectures)

Organic Reactions and Drug Synthesis (3L)

- Comparison of acidity and basicity of acids, alcohols, and amines
- Nucleophilic substitution and electrophilic addition reactions
- Markovnikov's rule, peroxide effect
- Synthesis and uses of Paracetamol and Aspirin (Name reactions not included)

Spectroscopy (2L)

- Electromagnetic spectrum, Lambert-Beer Law
- Finding of λ max value & concentration of the unknown solution,
- Applications of UV-VIS spectroscopy, Chromophores and Auxochromes
- Applications of IR spectroscopy, Fingerprint region

Text Books

1. *Chemistry – I*, Gourkrishna Das Mohapatro
2. *A Textbook of Engineering Chemistry*, Dr. Rajshree Khare
3. *Engineering Chemistry*, U. N. Dhar
4. *Physical Chemistry*, P. C. Rakshit

Reference Books

1. *Engineering Chemistry*, Jain & Jain
2. *Engineering Chemistry (NPTEL Web-book)*, B. L. Tembe, Kamaluddin and M. S. Krishna
3. *Textbook of Engineering Chemistry*, Jaya Shree Anireddy

Course Name: Engineering Mathematics - II

Course Code: M201

Contact (Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 3

Prerequisite:

The students to whom this course will be offered must have the understanding of (10+2) standard algebraic operations, coordinate geometry, and elementary calculus concepts including limits, continuity, differentiation, and integration.

Course Objective:

- Develop a thorough understanding of ordinary differential equations and their role in modeling real-world systems.
- Build competency in applying the Laplace transform as a tool for solving initial value problems and linear differential equations in engineering contexts.
- Gain proficiency in numerical techniques for solving mathematical problems where analytical methods are difficult or impossible.

Course Outcome(s):

After the completion of the course students will be able to

CO1: Apply analytical methods to solve ordinary differential equations in engineering contexts.

CO2: Apply the properties and inverse of Laplace Transforms to compute improper integrals and determine solutions of linear ordinary differential equations with constant coefficients in engineering scenarios.

CO3: Apply numerical methods to interpolate data, perform numerical integration, and solve ordinary differential equations in engineering applications.

CO4: Analyze the behavior of solutions using analytical and numerical approaches, including Laplace transforms, to assess stability, convergence, and accuracy in engineering contexts.

CO-PO-PSO Mapping

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

Course Content

Module I: First Order Ordinary Differential Equations (9L)

Solution of first order and first degree ODE: Exact ODE, rules for finding integrating factors, linear ODE, Bernoulli's equation.
Solution of first order and higher degree ODE: solvable for p, solvable for y and x, and Clairaut's equation.

Module II: Second Order Ordinary Differential Equations (8L)

Solution of second order ODE with constant coefficients: complementary function (CF), particular integral (PI), method of variation of parameters, and Cauchy-Euler equations.

Module III: Laplace Transform (12L)

Concept of improper integrals; definition and existence of Laplace Transform (LT); LT of elementary functions. First and second shifting properties, change of scale property, LT of $tf(t)$ and $f(t)t$, LT of derivatives and integrals of $f(t)$.
Evaluation of improper integrals using LT, LT of periodic and step functions. Inverse LT: definition and properties. Convolution theorem (statement only) and its application.
Solution of linear ODE with constant coefficients (initial value problem) using LT.

Module IV: Numerical Methods (7L)

Error analysis, calculus of finite differences.
Interpolation: Newton forward and backward, Lagrange's interpolation.
Numerical integration: trapezoidal rule, Simpson's $\frac{1}{3}$-rule.
Numerical solution of ODE: Euler method, fourth-order Runge-Kutta method.

Text Books

1. *Higher Engineering Mathematics*, B.S. Grewal, Khanna Publishers, 36th Edition, 2010.
2. *Advanced Engineering Mathematics*, E. Kreyszig, 9th Edition, John Wiley & Sons, 2006.

Reference Books

1. *A Textbook of Engineering Mathematics-I*, S. Guruprasad, New Age International Publishers.
2. *Higher Engineering Mathematics*, B.V. Ramana, Tata McGraw-Hill, 11th Reprint, 2010.
3. *Engineering Mathematics for First Year*, T. Veerarajan, Tata McGraw-Hill, 2008.
4. *A Textbook of Engineering Mathematics*, N.P. Bali and M. Goyal, Laxmi Publications, Reprint, 2008.
5. *Calculus and Analytic Geometry*, G.B. Thomas and R.L. Finney, 9th Edition, Pearson, 2002.
6. *Calculus (Vol. 1 and 2)*, M. Apostol, 2nd Edition, Wiley Eastern, 1980.
7. *Linear Algebra: A Geometric Approach*, S. Kumaresan, Prentice Hall of India, 2000.
8. *Linear Algebra: A Modern Introduction*, D. Poole, 2nd Edition, Brooks/Cole, 2005.
9. *Schaum's Outline of Matrix Operations*, R. Bronson, 1988.
10. *Differential and Integral Calculus (Vol. I & II)*, N. Piskunov, Mir Publishers, 1969.

Course Name: Design Thinking and Innovation

Course Code: HU203

Contact (Periods/Week): 1 periods

Total Contact Hours: 15

Credits: 1

Prerequisite:

For a course on the Basics of Design Thinking, students should ideally possess basic computer skills, communication abilities, problem-solving aptitude, critical thinking, introductory knowledge of Sustainable Development Goals, curiosity, and openness to new ideas, as well as basic understanding of mathematics, technology, and manufacturing processes. However, even if these prerequisites are not satisfied, the faculty will cover them in the first few classes. An awareness of 21st-century skills, including creativity and collaboration, is also beneficial. These prerequisites aim to provide a foundation, and any gaps in knowledge will be addressed by the instructor early in the course.

Course Objective:

The objective of this Course is to provide new ways of creative thinking and learn the innovation cycle of Design Thinking process for developing innovative products and services which are useful for a student in preparing for an engineering career.

Course Outcome(s):

After the completion of the course students will be able to

CO1: Analyze emotional experience and expressions to better understand stakeholders while designing innovative products through group brainstorming sessions.

CO2: Generate and develop design ideas through different technique.

CO3: Develop new ways of creative thinking and learn the innovation cycle of Design Thinking process for developing any innovative products using facility in AICTE IDEA LAB.

CO-PO-PSO Mapping

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

Course Content

Design Thinking Syllabus

Module	Content	Hours
Module 1	**Basics of Design Thinking**: Definition of Design Thinking, Need for Design Thinking, History of Design Thinking, Concepts & Brainstorming, 2 × 2 matrix, 6-3-5 method, NABC method.	2

Module 2	**Process of Design Thinking**: Understanding Design Thinking, Shared model in team-based design, Theory and practice, Global presentation signers, MVP or Prototyping. **Stages of Design Thinking Process (with examples)**:
	Empathize (Ask 5 Why, 5W+H, Stakeholder map, Empathy Map, Peer observation, Trend analysis);
	Define (Storytelling, Critical items diagram, Define success);
	Ideate (Brainstorming, 2x2 matrix, 6-3-5 method, NABC method);
	Prototype (Types of prototypes, Focused experiments, Exploration map, MVP);
	Test (Feedback capture grid, A/B testing).
		4

Module 3	**Tools for Design Thinking**: Real-time design interaction, Analysis, Empathy, Collaboration in digital space, Distributed design collaboration.	2
Module 4	**Design Thinking in IT**: Business process modeling, Agile in virtual collaboration, Scenario-based prototyping.	2
Module 5	**Strategic Innovation via Design Thinking**: Growth, Story- telling, Strategic foresight, Change and sense-making, Value re-definition, Humanization, Rapid prototyping, Creative culture, Standardization, Business model integration.	2
Module 6	**Problem Solving & Critical Thinking**: Introduction to TRIZ, SCAMPER, UI and UX; Mapping 17 Sustainable Development Goals (SDGs) in product/service design; Introduction to 21st Cen- tury Skill Set.	3
—	**Case Study & Project Report Submission**	—

Text Books

1. *Karmin Design Thinking* by Dr. Bala Ramadurai, Mudranik Technology Private Ltd. ISBN: 978-93-5419-010-0.

2. John R. Karsnitz, Stephen O'Brien and John P. Hutchinson, *Engineering Design*, Cengage Learning (International Edition), Second Edition, 2013.

3. Roger Martin, *The Design of Business: Why Design Thinking is the Next Competitive Advantage*, Harvard Business Press, 2009.

4. Hasso Plattner, Christoph Meinel and Larry Leifer (Eds.), *Design Thinking: Understand – Improve – Apply*, Springer, 2011.

5. Idris Mootee, *Design Thinking for Strategic Innovation: What They Can't Teach You at Business or Design School*, John Wiley & Sons, 2013.

Reference Books

1. Yousef Haik and Tamer M. Shahin, *Engineering Design Process*, Cengage Learning, Second Edition, 2011.
2. Jeanne Liedtka, Andrew King, Kevin Bennett, *Solving Problems with Design Thinking - Ten Stories of What Works*, Columbia Business School Publishing, 2013.
3. Tim Brown, *Change by Design: How Design Thinking Transforms Organizations and Inspires Innovation*, HarperCollins e-books, 2009.
4. Michael Lewrick, Patrick Link, Larry Leifer, *The Design Thinking Toolbox*, John Wiley & Sons, 2020.
5. Michael Lewrick, Patrick Link, Larry Leifer, *The Design Thinking Playbook*, John Wiley & Sons, 2018.
6. Kristin Fontichiaro, *Design Thinking*, Cherry Lake Publishing, USA, 2015.
7. Walter Brenner, Falk Uebernickel, *Design Thinking for Innovation - Research and Practice*, Springer Series, 2016.
8. Gavin Ambrose, Paul Harris, *Design Thinking*, AVA Publishing, 2010.
9. Muhammad Mashhood Alam, *Transforming an Idea into Business with Design Thinking*, First Edition, Taylor and Francis Group, 2019.
10. S. Balaram, *Thinking Design*, Sage Publications, 2011.

Web References

1. https://designthinking.ideo.com/
2. https://thinkibility.com/2018/12/01/engineering-vs-design-thinking/
3. https://www.coursera.org/learn/design-thinking-innovation
4. https://swayam.gov.in/nd1_noc20_mg38/preview
5. https://www.tutor2u.net/business/presentations/productlifecycle/default.html
6. https://docs.oracle.com/cd/E11108_02/otn/pdf/E11087_01.pdf
7. https://www.bizfilings.com/Home/Marketing/Product-Development
8. https://www.mindtools.com/brainstm.html
9. https://www.quicksprout.com/how-to-reverse-engineer-your-competit
10. https://www.vertabelo.com/blog/documentation/reverse-engineering
11. https://support.microsoft.com/en-us/kb/273814
12. https://support.google.com/docs/answer/179740?hl=en

Course Name: Constitution of India and Professional Ethics

Course Code: HU205

Contact (Periods/Week): 1 periods

Total Contact Hours: 12

Credits: 1

Prerequisite:

A basic knowledge (10+2 level) of the Indian Constitution and moral science.

Course Objective:

The objectives of this course are to make the student able to-

- Understand the salient features of the Indian constitution and form of government.
- Develop ethical awareness and responsible professional conduct.
- Understand ethical frameworks, guidelines and recognize ethical dilemmas.
- Understand professional responsibilities and applications of ethical principles in real-life scenarios.
- Develop an awareness of the social impact of the profession and act responsibly in the broader community.

Course Outcome(s):

After the completion of the course students will be able to

CO1: Identify, define and understand the significance of the Constitution of India, its spirit and values and the fundamental rights and duties as a responsible citizen.

CO2: Define and discover core ethical concepts, the basic perception of profession, and professional ethics that shape the ethical behavior of an engineer.

CO3: Identify, examine and apply codes of engineering ethics, engineers' social responsibilities and industrial standards and ethical dilemmas.

CO4: Consider, correlate and appraise ethical leadership and principles in addressing gender issues, concerns of IPR and industrial responsibilities.

CO-PO-PSO Mapping

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

Course Content

Module 1: Introduction to the Constitution of India and Indian Government (2L)

Preamble: Salient Features, Fundamental Rights, Fundamental Duties, Directive Principles of State Policy.
Parliament – Powers and Functions, Executive – President, Governor, Council of Ministers.

Module 2: Professional Ethics and Human Values (3L)

Introduction to Ethical Thinking; What is Ethics, Work Ethics.
Scope of Professional Ethics, Values and Characteristics.
Types of Values: Negative and Positive Values, Ethical Values for Professional Success.

Module 3: Codes of Professional Ethics, Violation and Safeguards (4L)

Engineering Ethics, Ethical Theories: Overview of Utilitarianism, Deontology, Virtue Ethics.
Professional Codes, Codes of Professional Ethics.
Moral Dilemmas and Moral Autonomy.
Internal Ethics of Business: Whistle Blowing, Conflicts of Interest, Job Discrimination, Exploitation of Employees.
Social and Ethical Responsibilities of Technologists: Responsibilities towards Customers, Shareholders, Employees.
Social Audit.
Case Studies: Bhopal Gas Tragedy, Chernobyl (Linking Ethics to Real-World Failures).

Module 4: Business Ethics and Workplace Issues (3L)

Business Ethics, Ethical Decision-Making Frameworks.
Impact of Ethics on Business Policies and Strategies.
Characteristics of Ethical Leaders; Fostering Integrity in Teams.
Addressing Occupational Crime, Discrimination, and Gender-Based Issues in Workplaces.
Intellectual Property Rights (IPR), Plagiarism, and Academic Misconduct.

Text Books

1. Durga Das Basu. *Introduction to the Constitution of India*, 27th ed., New Delhi: Lexis Nexis, 2024.
2. R. S. Naagarazan. *A Textbook on Professional Ethics and Human Values*, New Age International (P) Limited, 2022.
3. N. Subramanian. *Professional Ethics*, New Delhi: Oxford University Press, 2017.
4. A. N. Tripathi. *Human Values*, New Delhi: New Age Publishers, 2019.
5. S. K. Chakraborty. *Values and Ethics for Organizations: Theory and Practices*, New Delhi: Oxford University Press, 1997.

Reference Books

1. O. C. Ferrell, John Friaedrich, and Linda Ferrell. *Business Ethics: Ethical Decision Making and Cases*, New Delhi: Cengage India, 2024.

2. Charles Fledderman. *Engineering Ethics*, 3rd ed., New Delhi: Pearson Education, 2007.

3. Dinesh G. Harkut and Gajendra R. Bamnote. *Professional Ethics for Engineers*, Chennai: Notion Press, 2023.

4. U. C. Mathur. *Corporate Governance and Business Ethics: Text and Cases*, Chennai: Macmillan, 2012.

5. A. C. Fernando, K. P. Muralidheeran, and E. K. Satheesh. *Business Ethics – An Indian Perspective*, New Delhi: Pearson Education, 2019.

Course Name: Data Structures and Algorithms Lab

Course Code: CS291

Contact (Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 1.5

Prerequisite:

A basic knowledge programing language.

Course Objective:

By the end of this course, students will be able to:

- To develop skills in implementing and analyzing data structures using C.
- To gain hands-on experience in solving problems using arrays, linked lists, stacks, queues, trees, graphs, and hashing.
- To apply algorithmic concepts like recursion, sorting, and searching in solving real-world problems.

Course Outcome(s):

After the completion of the course students will be able to

CO1: Apply fundamental programming concepts to construct and manipulate linear data structures like arrays, linked list, stacks, and queues for solving structured problems.

CO2: Develop and analyze non-linear data structures such as binary search trees to address hierarchical and dynamic memory-based problems.

CO3: Implement recursive algorithms to solve classical problems like Tower of Hanoi and Fibonacci series, demonstrating critical thinking and abstraction.

CO4: Compare and evaluate various sorting and searching algorithms based on time and space complexity for performance-critical applications.

CO5: Design and integrate suitable data structures to build efficient software modules, demonstrating teamwork, project planning, and communication of technical results.

CO-PO-PSO Mapping

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

Course Content

Lab No.	Title	Topics / Experiments
1	Introduction to C Revisions	Basic C programming constructs, functions, pointer concepts.
2	Arrays and Polynomial Representation	Create, access and manipulate 1D, 2D arrays; polynomial representation using arrays.
3	Linked Lists	Singly Linked List: creation, insertion, deletion, search.
4	Doubly & Circular Linked Lists	Implement doubly linked and circular linked list with insertion/deletion.
5	Stacks (Array & Linked List)	Implement stack using array and linked list.
6	Application of Stack	Infix to postfix conversion, postfix evaluation.
7	Queues (Array & Linked List)	Physical, Linear and Circular model of queues using array, Queue using linked list.
8	Recursion Applications	Factorial, Fibonacci, Tower of Hanoi.
9	Binary Search Tree (BST)	Insertion, deletion, searching; height of tree.
10	Sorting Algorithms	Implementation of bubble sort, insertion sort, and selection sort.
11	Sorting Algorithms	Implementation of quick, merge sort, and radix sort.
12	Searching and Hashing	Linear search, binary search, interpolation search.

Text Books

1. Samiran Chattopadhyay, Debabrata Ghosh Dastidar, Matangini Chattopadhyay, *Data Structures Through 'C' Language*, Edition: 2001, BPB Publications.
2. Ellis Horowitz, Sartaj Sahni, Susan Anderson-Freed, *Fundamentals of Data Structures in C*, 2nd Edition, Universities Press.

Reference Books

1. Thomas A. Standish, *Data Structures, Algorithms, and Software Principles in C*, 1st Edition, Pearson.
2. Seymour Lipschutz, *Data Structures*, Special Indian Edition, Tata McGraw Hill Education.
3. Robert L. Kruse, Bruce P. Leung, *Data Structures and Program Design in C*, 2nd Edition, Pearson.
4. Aaron M. Tenenbaum, *Data Structures in C*, 1st Edition, Pearson.

Course Name: Introduction to Artificial Intelligence Lab

Course Code: CS292

Contact (Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 1.5

Prerequisite:

Basic Computer Knowledge.

Course Objective:

By the end of this course, students will be able to:

- Gain foundational knowledge of PROLOG to implement an Artificial Intelligent Agent as an executable computer program for Knowledge Representation and Inferencing.
- Formulate a problem by analysing its characteristics to fit a State-Space Exploration Framework or an Inferencing Framework of Artificial Intelligence.
- Apply the concepts of Artificial Intelligence to solve a problem by implementing well-known Artificial Intelligence strategies using proper techniques and tools of PROLOG.
- Build expert systems offering solutions to the challenging problems of Artificial Intelligence.
- Implement Artificial Intelligence based ideas as executable PROLOG programs through developing intelligent heuristic strategies.

Course Outcome(s):

After the completion of the course students will be able to

CO1: Acquire foundational knowledge of PROLOG to implement an Artificial Intelligent Agent as an executable computer program for Knowledge Representation and Inferencing and understand the working principle of the agent and assess its utilitarian importance in current technological context leading towards lifelong learning.

CO2: Identify and formulate an engineering problem by analyzing its characteristics to fit a State-Space Exploration Framework or an Inferencing Agent Formulation Framework of Artificial Intelligence.

CO3:Explore relevant literature and apply the concepts of Artificial Intelligence to solve a problem by implementing well-known Artificial Intelligence strategies using proper techniques and tools of PROLOG.

CO4: Develop ideas and propose an expert system offering solutions to the challenging problems of Artificial Intelligence.

CO5: Plan and Implement Artificial Intelligence based ideas as executable PROLOG programs through developing intelligent heuristic strategies or expert systems with adequate documentation in a collaborative environment for successfully carrying out projects on Artificial Intelligence Problems and investigate their effectiveness by analysing the performances using proper techniques and tools.

CO-PO-PSO Mapping

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

Course Content

Mod.	Content
1	**Introduction to PROLOG Programming and IDE Basics** (3 Lectures) Assignments for understanding the basic components of knowledge representation and inferencing in Artificial Intelligence using PROLOG Programming and its working strategy. Topics include understanding facts, rules, queries, and syntax.
2	**Recursive Definitions in Prolog** (5 Lectures) Implement Fibonacci Series, calculator functions, factorial computation, summation, list length, etc., using recursive rules.
3	**Defining Facts and Simple Queries** (4 Lectures) Writing a knowledge base for family relationships and basic objects.
4	**Rules and Inference in Prolog** (4 Lectures) Creating logical rules and testing inferences with example queries.
5	**List Operations in Prolog** (4 Lectures) Operations including membership checking, list concatenation, reversal, and finding maximum/minimum of a list.
6	**Pattern Matching and Symbolic Reasoning** (5 Lectures) Simple examples involving pattern recognition (e.g., shape or name matching), family tree design and symbolic inference.
7	**Expert System Simulation (Mini Project)** (5 Lectures) Building a mini knowledge-based expert system (e.g., Animal Classification, Medical Diagnosis).

Text Book

1. Ivan Bratko, *Prolog Programming for Artificial Intelligence*, 4th Edition, Addison-Wesley.

Course Name: Digital Logic and Computer Organization Lab

Course Code: CS293

Contact (Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 1.5

Prerequisite:

Basic Computer Knowledge.

Course Objective:

By the end of this course, students will be able to:

- To provide hands-on experience in designing and analyzing combinational and sequential logic circuits.
- To enhance understanding of digital systems using simulation and HDL tools.
- To familiarize students with arithmetic circuits, memory design, and basic CPU control logic through practical implementation.

Course Outcome(s):

After the completion of the course students will be able to

CO1: Implement and verify the functionality of basic and derived logic gates, using ICs and simulation tools to demonstrate fundamental digital operations.

CO2: Design and simplify combinational logic circuits from Boolean expressions using Karnaugh Maps, and simulate them for correctness and efficiency.

CO3: Construct and analyze sequential circuits such as flip-flops, counters, and shift registers to demonstrate state behavior and timing sequences.

CO4: Develop arithmetic circuits and evaluate algorithmic performance (e.g., Booth's multiplication) using Hardware Description Languages (HDL).

CO5: Integrate combinational and sequential modules to create a simplified CPU architecture through collaborative mini-projects, enhancing teamwork, communication, and project management skills.

CO-PO-PSO Mapping

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

Course Content

Lab No.	Title	Description
1	Basic Logic Gates	Implement and verify truth tables of NOT, AND, OR, NAND, NOR, XOR, XNOR gates using ICs and simulation software.
2	Boolean Expression Simplification	Design logic circuits from Boolean expressions, simplify using Karnaugh Maps, and simulate the simplified circuit.
3	Combinational Circuit – Adders & Subtractors	Implement Half-Adder, Full-Adder, Half-Subtractor, and Full-Subtractor using logic gates and ICs.
4	Design of Code Converters	Design and implement Binary to Gray, Gray to Binary, Binary to BCD, and BCD to Excess-3 converters.
5	Multiplexers and Demultiplexers	Design and verify 4:1, 8:1 MUX and 1:4, 1:8 DEMUX using logic gates and ICs or simulation tools.
6	Encoders and Decoders	Implement 8-to-3 encoder and 3-to-8 decoder using logic gates and analyze their truth tables.
7	Flip-Flops and Latches	Design and test SR, JK, D, T flip-flops using ICs or HDL; study race-around and master-slave configurations.
8	Synchronous and Asynchronous Counters	Design and simulate up/down counters (binary, mod-n, ring, Johnson) and study their timing behavior.
9	Shift Registers	Implement SISO, SIPO, PIPO, and PISO registers using flip-flops or HDL; demonstrate serial and parallel operations.
10	Arithmetic Circuits Using HDL	Model addition, subtraction, Booth's multiplication, restoring and non-restoring division using Verilog/VHDL.
11	Memory and Address Decoding	Simulate basic RAM/ROM using HDL and design address decoder circuits for memory mapping.
12	Mini Project / CPU Module Simulation	Group-based implementation of a simple CPU datapath (ALU + Register File + Control Unit) using HDL or simulation.

Tools and Resources

Software: Logisim, Multisim, ModelSim, Xilinx Vivado / ISE, Quartus
Hardware Kits: Digital Trainer Kit, ICs (74xx series), LEDs, switches, Breadboards
Languages: Verilog/VHDL (optional for advanced simulation)

Textbooks

1. M. Morris Mano, *Digital Logic and Computer Design*, 1st Edition, Pearson Education.
2. William Stallings, *Computer Organization and Architecture: Designing for Performance*, 10th Edition, Pearson Education.

Reference Books

1. M. Morris Mano and Michael D. Ciletti, *Digital Design*, 5th Edition, Pearson Education.
2. Carl Hamacher, Zvonko Vranesic, Safwat Zaky, *Computer Organization and Embedded Systems*, 6th Edition, McGraw-Hill Education.
3. David A. Patterson, John L. Hennessy, *Computer Organization and Design: The Hardware/Software Interface*, RISC-V Edition, Morgan Kaufmann Publishers.
4. Charles H. Roth Jr., Larry L. Kinney, *Fundamentals of Logic Design*, 7th Edition, Cengage Learning.
5. Thomas L. Floyd, *Digital Fundamentals*, 11th Edition, Pearson Education.

Course Name: Engineering Chemistry Lab

Course Code: CH291

Contact (Periods/Week): 2 periods

Total Contact Hours: 24

Credits: 1

Prerequisite:

10+2

Course Objective:

By the end of this course, students will be able to:

- Study the basic principles of pH meter and conductivity meter for different applications
- Analysis of water for its various parameters in relation to public health, industries & environment
- Learn to synthesis Polymeric materials and drugs.
- Study the various reactions in homogeneous and heterogeneous medium.
- Designing of innovative experiments.

Course Outcome(s):

After the completion of the course students will be able to

CO1: Able to operate different types of instruments for estimation of small quantities chemicals used in industries and scientific and technical fields.

CO2: Able to analyse and determine the composition and physical property of liquid and solid samples when working as an individual and also as a team member.

CO3: Able to analyse different parameters of water considering environmental issues.

CO4: Able to synthesize drug and sustainable polymer materials.

CO5: Capable to design innovative experiments applying the fundamentals of modern chemistry.

CO-PO-PSO Mapping

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

Course Content

List of Chemistry Laboratory Experiments

1. **Estimation of Sodium Hydroxide:**
   To determine the strength of a given sodium hydroxide solution by titrating against standard oxalic acid solution.
2. **Estimation of Iron (II):**
   Estimation of amount of Fe^{2+} in Mohr's salt using permanganometry.
3. **Surface Tension Measurement:**
   To determine the surface tension of a given liquid at room temperature using a stalagmometer by the drop number method.
4. **Viscosity Measurement:**
   To determine the viscosity of a given unknown liquid with respect to water at room temperature using Ostwald's viscometer.
5. **Water Quality Analysis: Hardness**
   Determination of total, permanent, and temporary hardness of a water sample by complexometric titration using EDTA.
6. **Water Quality Analysis: Chloride Ion Concentration**
   Determination of chloride ion concentration in a water sample by argentometric titration method.
7. **pH-Metric Titration:**
   Determination of the strength of a given HCl solution by titration against a standard NaOH solution using a pH meter.
8. **Conductometric Titration:**
   Determination of the strength of a given HCl solution using conductometric titration against a standard NaOH solution.
9. **Partition Coefficient:**
   Determination of the partition coefficient of acetic acid between water and butanol (two immiscible liquids).
10. **Synthesis of Bakelite Polymer:**
    Preparation of Bakelite polymer, used in electrical insulators and printed circuit boards (PCBs).

Course Name: IDEA LAB Workshop

Course Code: ME292

Contact (Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 1.5

Course Objective:

- To learn all the skills associated with the tools and inventory associated with the IDEA Lab.
- Learn useful mechanical and electronic fabrication processes.
- Learn necessary skills to build useful and standalone system/ project with enclosures.
- Learn necessary skills to create print and electronic documentation for the system/project.

Course Contents

Module-wise Topics

Module	Topics
1	Electronic component familiarisation, Understanding electronic system design flow. Schematic design and PCB layout and Gerber creation using EagleCAD. Documentation using Doxygen, Google Docs, Overleaf. Version control tools - GIT and GitHub. Basic 2D and 3D designing using CAD tools such as FreeCAD, Sketchup, Prusa Slicer, FlatCAM, Inkspace, OpenBSP and VeriCUT. Introduction to basic hand tools - Tape measure, combination square, Vernier calliper, hammers, fasteners, wrenches, pliers, saws, tube cutter, chisels, vice and clamps, tapping and threading. Adhesives. Introduction to Power tools: Power saws, band saw, jigsaw, angle grinder, belt sander, bench grinder, rotary tools. Various types of drill bits.
2	Familiarisation and use of basic measurement instruments - DSO including various triggering modes, DSO probes, DMM, LCR bridge, Signal and function generator. Logic analyzer and MSO. Bench power supply (with 4-wire output). Circuit prototyping using (a) breadboard, (b) Zero PCB, (c) 'Manhattan' style and (d) custom PCB. Single, double and multilayer PCBs. Single and double-sided PCB prototype fabrication in the lab. Soldering using soldering iron/station. Soldering using a temperature controlled reflow oven. Automated circuit assembly and soldering using pick and place machines. Mechanical cutting processes - 3-axis CNC routing, basic turning, milling, drilling and grinding operations, Laser cutting, Laser engraving etc. Basic welding and brazing and other joining techniques for assembly. Concept of Lab aboard a Box.
3	Electronic circuit building blocks including common sensors. Arduino and Raspberry Pi programming and use. Digital Input and output. Measuring time and events. PWM. Serial communication. Analog input. Interrupts programming. Power Supply design (Linear and Switching types), Wireless power supply, USB PD, Solar panels, Battery types and charging. 3D printing and prototyping technology – 3D printing using FDM, SLS and SLA. Basics of 3D scanning, point cloud data generation for reverse engineering. Prototyping using subtractive cutting processes. 2D and 3D Structures for prototype building using Laser cutter and CNC routers. Basics of IPR and patents; Accessing and utilizing patent information in IDEA Lab.
4	Discussion and implementation of a mini project.

5	Documentation of the mini project (Report and video).

Laboratory Activities

S. No.	List of Lab Activities and Experiments
1	Schematic and PCB layout design of a suitable circuit, fabrication and testing of the circuit.
2	Machining of 3D geometry on soft material such as soft wood or modelling wax.
3	3D scanning of computer mouse geometry surface. 3D printing of scanned geometry using FDM or SLA printer.
4	2D profile cutting of press fit box/casing in acrylic (3 or 6 mm thickness)/cardboard, MDF (2 mm) board using laser cutter and engraver.
5	2D profile cutting on plywood /MDF (6-12 mm) for press fit designs.
6	Familiarity and use of welding equipment.
7	Familiarity and use of normal and wood lathe.
8	Embedded programming using Arduino and/or Raspberry Pi.
9	Design and implementation of a capstone project involving embedded hardware, software and machined or 3D printed enclosure.

Reference Books

S. No.	Title
1	AICTE's Prescribed Textbook: Workshop / Manufacturing Practices (with Lab Manual), Khanna Book Publishing, New Delhi.
2	All-in-One Electronics Simplified, A.K. Maini; 2021. ISBN-13: 978-9386173393, Khanna Book Publishing Company, New Delhi.
3	Simplified Q&A - Data Science with Artificial Intelligence, Machine Learning and Deep Learning, Rajiv Chopra, ISBN: 978-9355380821, Khanna Book Publishing Company, New Delhi.
4	3D Printing & Design, Dr. Sabrie Soloman, ISBN: 978-9386173768, Khanna Book Publishing Company, New Delhi.
5	The Big Book of Maker Skills: Tools & Techniques for Building Great Tech Projects. Chris Hackett. Weldon Owen; 2018. ISBN-13: 978-1681884325.
6	The Total Inventors Manual (Popular Science): Transform Your Idea into a Top-Selling Product. Sean Michael Ragan (Author). Weldon Owen; 2017. ISBN-13: 978-1681881584.
7	Make: Tools: How They Work and How to Use Them. Platt, Charles. Shroff/Maker Media. 2018. ISBN-13: 978-9352137374.
8	The Art of Electronics. 3rd edition. Paul Horowitz and Winfield Hill. Cambridge University Press. ISBN: 9780521809269.

9	Practical Electronics for Inventors. 4th edition. Paul Sherz and Simon Monk. McGraw Hill. ISBN-13: 978-1259587542.
10	Encyclopedia of Electronic Components (Volume 1, 2 and 3). Charles Platt. Shroff Publishers. ISBN-13: 978-9352131945, 978-9352131952, 978-9352133703.
11	Building Scientific Apparatus. 4th edition. John H. Moore, Christopher C. Davis, Michael A. Coplan and Sandra C. Greer. Cambridge University Press. ISBN-13: 978-0521878586.
12	Programming Arduino: Getting Started with Sketches. 2nd edition. Simon Monk. McGraw Hill. ISBN-13: 978-1259641633.
13	Make Your Own PCBs with EAGLE: From Schematic Designs to Finished Boards. Simon Monk and Duncan Amos. McGraw Hill Education. ISBN-13: 978-1260019193.
14	Pro GIT. 2nd edition. Scott Chacon and Ben Straub. A press. ISBN-13: 978-1484200773.
15	Venuvinod, PK., MA. W., Rapid Prototyping – Laser Based and Other Technologies, Kluwer, 2004.
16	Ian Gibson, David W Rosen, Brent Stucker., “Additive Manufacturing Technologies: Rapid Prototyping to Direct Digital Manufacturing”, Springer, 2010.
17	Chapman W.A.J, “Workshop Technology”, Volume I, II, III, CBS Publishers and distributors, 5th Edition, 2002.

2nd Year 3rd Semester									
Sl. No.	Broad Catagory	Catagory	Paper Code	Subject	L	T	P	Total	Credit Points
A. Theory									
1	ENGG	Major	CB301	Design Analysis of Algorithms	3	0	0	3	3
2	ENGG	Major	CB302	Object-Oriented Programming Using C++	3	0	0	3	3
3	ENGG	Major	CB303	Database Management Systems	3	0	0	3	3
4	ENGG	Major	CB304	Computer Architecture	3	0	0	3	3
5	HUM	Multi	BS301	Marketing Management	2	0	0	2	2
6	SCI	Multi	M(CB)301	Statistical Methods	3	0	0	3	3
B. Practical									
1	ENGG	Major	CB391	Design Analysis of Algorithms Lab	0	0	3	3	1.5
2	ENGG	Major	CB392	Object-Oriented Programming Using C++ Lab	0	0	3	3	1.5
3	ENGG	Major	CB393	Database Management Systems Lab	0	0	3	3	1.5
4	ENGG	Major	CB394	Introduction to Java Lab	0	0	3	3	1.5
C. MANDATORY ACTIVITIES / COURSES									
1	MC	Mandatory Course	MC381	NSS/ Physical Activities/Meditation & Yoga/Photography or Nature Club	0	0	0	0	2 Units
Total of Theory, Practical								29	23

Course Name: Design and Analysis of Algorithms

Course Code: CB301

Contact (Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 3

Prerequisite:

Data Structures and Algorithms, Discrete Mathematics, Programming in C/C++

Course Objective:

By the end of this course, students will be able to:

- Understand the fundamental principles and techniques used in algorithm design and analysis.
- Analyze time and space complexities of algorithms using asymptotic notations.
- Apply mathematical strategies such as divide and conquer, greedy, and dynamic programming to solve computational problems.
- Evaluate algorithmic performance and prove correctness through formal methods.
- Explore NP-completeness and introduce the concept of intractable problems.

Course Outcome(s):

After the completion of the course students will be able to

CO1: Explain the fundamentals of algorithm design and illustrate the concept of computational complexity using engineering knowledge of mathematics and computing principles to analyze complex engineering problems.

CO2: Apply divide-and-conquer and recursive strategies and analyze their correctness and efficiency to develop solutions for complex problems, while ensuring sustainable and optimal performance.

CO3: Design optimized algorithms using greedy and dynamic programming paradigms and compare alternative solutions with regard to resource constraints and system-level efficiency, using modern engineering tools.

CO4: Develop solutions for combinatorial problems using backtracking and branch-and-bound and evaluate their performance using modeling, experimentation, and data analysis in a research-based approach.

CO5: Interpret complexity classes (P, NP, NP-Complete) and justify computational limitations in light of ethics, professional responsibility, and the need for lifelong learning and adaptability to emerging technolo- gies.

CO-PO-PSO Mapping

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

Course Content

Module 1:

- What is an Algorithm? Characteristics
- RAM Model, Algorithm Specification
- Asymptotic Notations: Big-O, Ω, Θ
- Time and Space Complexity Analysis
- Best, Worst, and Average Case Analysis

Module 2:

Divide and Conquer (5L)

- General Method, Recurrence Relations
- Merge Sort, Quick Sort
- Matrix Multiplication (Strassen’s Algorithm)
- Master Theorem and its Applications

Module 3:

Greedy Method & Disjoint Set Union–Find (6-L)

- General Strategy, Greedy vs. Dynamic Programming
- Huffman Coding
- Fractional Knapsack, Job Sequencing with Deadlines
- Disjoint Set Data Structures:
  - Union by Rank
  - Path Compression
- Prim’s and Kruskal’s Algorithms for MST

Module 4:

Dynamic Programming (6 L)

- Basic Concepts, Principle of Optimality
- 0/1 Knapsack Problem
- Longest Common Subsequence (LCS)
- Matrix Chain Multiplication
- Floyd–Warshall and Bellman–Ford Algorithms

Module 5:

Backtracking and Branch & Bound (3-L)

- *N*-Queens Problem
- Graph Coloring
- Hamiltonian Cycle

Module 6:

Graph Algorithms and Traversals (4-L)

- DFS and BFS
- Topological Sorting
- Connected Components
- Shortest Path: Dijkstra's Algorithm

Module 7:

NP-Completeness and Approximation Algorithms (7-L)

- *P*, *NP*, NP-Complete, NP-Hard Definitions
- Polynomial Reduction
- SAT, 3-SAT, Clique, Vertex Cover
- Approximation Algorithms:
  - Vertex Cover
  - Metric TSP (Triangle Inequality)
  - Set Cover
- Performance Guarantees and Approximation Ratios
- Implications of Intractability

Text Books

1. *Introduction to Algorithms* by T.H. Cormen, C.E. Leiserson, R.L. Rivest, and C. Stein, MIT Press, 3rd Edition.
2. *Algorithms* by S. Dasgupta, C.H. Papadimitriou, and U.V. Vazirani, McGraw Hill Education.
3. *Fundamentals of Computer Algorithms* by S. Sahni and E. Horowitz, Galgotia Publications.

Reference Book

1. *The Design and Analysis of Computer Algorithms* by A.V. Aho, J.E. Hopcroft, and J.D. Ullman, Pearson.

Course Name: Object-Oriented Programming Using C++

Course Code: CB302

Contact (Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 3

Prerequisite:

- Basic knowledge of C programming
- Understanding of problem-solving techniques and algorithms

Course Objective:

The objectives of this course are to:

1. Understand the fundamental concepts of object-oriented programming.
2. Apply the principles of abstraction, encapsulation, inheritance, and polymorphism in C++.
3. Develop problem-solving skills using classes, objects, and operator overloading.
4. Implement file handling and exception handling in C++.
5. Design and implement C++ programs for real-world applications.

Course Outcome(s):

After the completion of the course students will be able to

CO1: Explain the features and benefits of object-oriented programming over procedural programming.

CO2: Apply concepts of classes, objects, and constructors for modular program design.

CO3: Implement inheritance and polymorphism to achieve reusability and flexibility in programming.

CO4: Use operator overloading, templates, and exception handling to develop robust applications.

CO5: Develop complete C++ programs involving file handling and demonstrate debugging skills.

CO-PO-PSO Mapping

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

Course Content

Module 1:

Introduction to Object-Oriented Programming (6 L):
Procedural vs Object-Oriented Programming Basic concepts: Object, Class, Data abstraction, Encapsulation, Inheritance, Polymorphism Structure of a C++ program, Tokens, Data types, Operators, Control structures

Module 2:

Classes and Objects (8L):
Defining classes, creating objects Member functions, access specifiers Constructors and destructors Static members and friend functions

Module 3:

Inheritance and Polymorphism (8L):
Types of inheritance: single, multiple, multilevel, hierarchical, hybrid Virtual base classes, ambiguity reso- lution Function overloading, operator overloading Virtual functions and runtime polymorphism

Module 4:

Advanced Features :
Templates (function and class templates) Exception handling in C++ Introduction to Standard Template Library (STL): vector, list, stack, queue

Module 5:

File Handling and Applications (7L) File streams, opening/closing files Reading/writing text and binary files Error handling during file operations Mini-project: building a C++ application integrating OOP features

Text Books:

1. **”Object-Oriented Programming with C++”**, E. Balagurusamy, McGraw Hill.
2. **”The C++ Programming Language”**, Bjarne Stroustrup, Addison-Wesley, 4th Edition.

Reference Books:

1. **”Programming in C++”**,Ashok N. Kamthane, Pearson Education, 2nd Edition.
2. **”C++ Primer”**, Stanley B. Lippman, Jos´ee Lajoie, Barbara E. Moo, Addison-Wesley, 5th Edition.
3. **”Object-Oriented Programming in C++”**, Robert Lafore, Pearson Education, 4th Edition

Course Name: Database Management Systems

Course Code: CB303

Contact (Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 3

Prerequisite:

Logic of Programming Language, Basic Concepts of Data Structures and Algorithms

Course Objective:

- Understand the core principles of database systems, including architecture and data models.
- Develop the ability to model real-life scenarios using the ER model and convert them into relational models.
- Apply structured query language (SQL) to define and manipulate data in relational databases.
- Analyze relational schema using functional dependencies and normalize them for consistency.
- Demonstrate knowledge of transaction management, indexing, and file organization mechanisms in DBMS.

Course Outcome(s):

After the completion of the course students will be able to

CO1: Understand Database Management System and explain fundamental elements of a DBMS.

CO2: Compare relational data model, E-R model, and file organization; use appropriate indexing.

CO3: Apply efficient query optimization, transaction management, concurrency control, and recovery.

CO4: Analyze and improve database design using normalization techniques.

CO5: Design E-R diagrams, translate them to relational schemas, populate and query relational databases.

CO-PO-PSO Mapping

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

Course Content

Module 1:

Introduction [2L]
Concept & 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 Subqueries, Database Security, Application Development using SQL, Stored Procedures and Triggers.

Module 4:

Relational Database Design [8L]
Functional Dependency, Anomalies in Designing a Database, Normalization using Functional Dependencies, Decomposition, BCNF, 3NF, Multivalued Dependencies, 4NF, 5NF, Case Study.

Module 5:

Internals of RDBMS [7L]
Physical Data Structures, Query Optimization: Join Algorithm, Statistics and Cost-Based Optimization. Transaction Processing, Concurrency Control and Recovery Management: Transaction Model, Serializability, Locking Protocols, Two-Phase Locking, Deadlock Handling.

Module 6:

File Organization & Index Structures [6L]
File & Record Concepts, File Placement, Fixed and Variable Size Records, Single-Level Indexing (Primary, Secondary, Clustering), Multi-level Indexes.

Text Books:

1. **”Database System Concepts”**, Henry F. Korth and Abraham Silberschatz, McGraw Hill.
2. **”Fundamentals of Database Systems”**, Ramez Elmasri and Shamkant Navathe, Benjamin Cummings.

Reference Books:

1. **”Fundamentals of Database Systems”**, Ramez Elmasri and Shamkant Navathe, Addison-Wesley..
2. **”Database Management System”**, Raghu Ramakrishnan, McGraw-Hill.
3. **”Database Systems”**, Connolly and Begg, Pearson Education.

Course Name: Computer Architecture

Course Code: CB304

Contact (Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 3

Prerequisite:

Concept of basic components of a digital computer, Fundamentals & Program structures.

Course Objective:

- Understand the organization and architecture of basic computer systems.
- Apply knowledge of ALU design, memory architecture, and control unit design.
- Analyze pipelining and instruction-level parallelism techniques.
- Evaluate performance issues related to memory and processing units.
- Design small-scale processor components and evaluate their performance.

Course Outcome(s):

After the completion of the course students will be able to

CO1: Apply the operational concepts for instruction execution, arithmetic operations, control signals, memory operations and data transfer methods on various problems.

CO2: Analyze types of addressing modes, interrupts, arithmetic and logic circuits, memory and bus architectures and its timing diagrams.

CO3: Summarize the basic concept of pipeline, instruction pipeline, arithmetic pipeline hazards detection and prevention and use this knowledge for designing and implementing mathematical and engineering problems.

CO4: Develop technological aspects on computer organization and architecture to solve complex problems. CO5:

Evaluate and compare RISC and CISC architectures, memory hierarchy, and I/O processing.

CO-PO-PSO Mapping

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

Course Content

Module – 1: [6L]

Basic organization of the stored program computer and operation sequence for execution of a program. Role of operating systems and compiler/assembler. Fetch, decode and execute cycle, Concept of operator, operand, registers and storage, Instruction format. Instruction sets and addressing modes.

Module 2: [6L]

Overflow and underflow. Design of adders - ripple carry and carry look ahead principles. Design of ALU. Fixed point multiplication -Booth's algorithm. Fixed point division - Restoring and non-restoring algorithms. Floating point - IEEE 754 standard.

Module 3: [6L]

Memory unit design with special emphasis on implementation of CPU-memory interfacing. Memory organization, static and dynamic memory, memory hierarchy, associative memory. Inclusion, Coherence and locality properties; Cache memory organizations, Techniques for reducing cache misses; Virtual memory organization, mapping and management techniques, memory replacement policies.

Module 4: [6L]

Design of control unit - hardwired and micro programmed control. Introduction to RISC architectures. RISC vs CISC architectures. I/O operations - Concept of handshaking, Polled I/O, interrupt and DMA.

Module 5: [10L]

Pipelining: Basic concepts, instruction and arithmetic pipeline[2L], data hazards, control hazards and structural hazards, techniques for handling hazards[2L] Pipeline vs. Parallelism, Levels of parallelism [1L], Instruction-Level Parallelism: Basic Concepts, Techniques for Increasing ILP, Superscalar, Super Pipelined and VLIW Processor Architectures [3L], Array and Vector Processors [2L]

Module 6: [2L]

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

Text Books:

1. **“Computer System Architecture”**,Mano, M.M., PHI.
2. **”Advance Computer Architecture”**, Kai Hwang, McGraw Hill.
3. **” Computer Architecture”**,Behrooz Parham, Oxford University Press.
4. **”Computer Architecture & Organization”**, Nicholas P Carter, McGraw-Hill.

Reference Books:

1. **“Computer Architecture & Organisation”**,Hayes J. P, McGraw Hill
2. **”Computer Organisation”**, Hamacher, McGraw Hill.
3. **” Microprocessors and Microcontrollers”**,N. senthil Kumar, M. Saravanan, S. Jeevananthan.
4. **”Computer Organisation & Design”**, Chaudhuri P. Pal, PHI.
5. **”Computer Organization & Architecture”**, P N Basu, Vikas Pub.

Course Name: Marketing Management

Course Code: BS301

Contact (Periods/Week): 2 periods

Total Contact Hours: 24

Credits: 2

Prerequisites: None

Course Objective(s):
The objective of the course is to make the students able to –

1. Orient engineering students towards being strategic decision makers in marketing of ideas, products, and services apart from technical knowledge in the dynamic market environment.
2. Introduce the basic concepts of marketing, consumer behavior, and marketing strategies with respect to product, price, place and promotion.
3. Provide insights into new product development, pricing strategies, selection of channels for distribution, and promotion of products.

Course Outcomes (COs): After successful completion of the course, students will be able to:

CO1: Understand the concepts of marketing, factors influencing consumer behavior, decision-making process, and strategic areas of 4Ps.

CO2: Understand the insight earned about consumer psychology in improving the demand of the product in the market.

CO3: Analyze markets and consumers, the changing environmental factors with special focus on technology products.

CO4: Create an appropriate strategy for the marketing of high-tech products and services.

CO5: Develop new products and services that are consistent with evolving marketing needs.

CO-PO Mapping: Course Content:

CO-PO-PSO Mapping

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

1. **Introduction to Marketing (04L)**
   Evolution of Marketing Concept – Core concepts of marketing – Scope and Importance of Marketing – Difference between Selling and Marketing – Marketing Myopia – Consumer Marketing vs. Industrial Marketing.

2. **Consumer Behaviour (06L)**
   Understanding Consumer Behaviour: nature, scope and importance – Factors influencing Consumer Behavior – Buying decision-making process – Market Segmentation, Targeting and Positioning (STP).

3. **Product (04L)**
   Marketing mix – Product definition, levels of product, product classification, difference between goods and services – Product Life Cycle – New Product Development – Technology and Product Management.

4. **Pricing and Physical Distribution (05L)**
   Price – Pricing Objectives – Pricing Policies – Methods of Pricing – Distribution Channel (Levels, Advantages and Disadvantages) – Factors to be considered in Selecting a Channel – Channel Conflicts (Causes and Overcoming Conflicts).

5. **Promotion (05L)**
   Promotion mix – Marketing Communication Tools for High-Tech Markets – Channels of distribution – Supply Chain Management in High-Tech Markets – Technology Marketing – Green Marketing – Introduction to market study.

Text Books:

1. Dr. C. B. Gupta, Dr. N. Rajan Nair, *Marketing Management*, Sultan Chand & Sons, New Delhi.
2. Philip Kotler, *Marketing Management*, Prentice Hall of India Pvt. Ltd., New Delhi.

Reference Books:

1. R. S. N. Pillai & Bagavathi, *Marketing Management*, S. Chand Publishing.
2. Rajan Saxena, *Marketing Management*, McGraw Hill Education.
3. V. S. Ramaswamy, S. Namakumari, *Marketing Management: Global Perspective*, Sage Publications India Pvt. Ltd., New Delhi.

Course Name: Statistical Method

Course Code: M(CB)301

Contact:3

Total Contact Hours: 36 Hours

Credits: 3

Prerequisites:

Basic understanding of algebra and calculus.

Course Objective(s):

The objective of the course is to make the students able to –

- O1: Understand basic probability theory and random variables.
- O2: Apply knowledge of probability distributions and expectations in real-world contexts.
- O3: Summarize and interpret data using descriptive and inferential statistics.
- O4: Develop estimators and perform hypothesis testing.
- O5: Apply statistical models like regression, correlation, and ANOVA to data.

Course Outcomes (COs):

After successful completion of the course, students will be able to:

CO1: Understand the concept of basic probability theory and random variables.

CO2: Compute probability distributions and expectations in real-world contexts.

CO3: Analyze and summarize descriptive and inferential statistics.

CO4: Apply sampling techniques, point estimation and hypothesis testing procedures.

CO5: Evaluate regression, correlation, and ANOVA models for statistical analysis.

CO-PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	0	0	0	0	0	0	0	0	0	0	0	0
CO2	3	3	2	0	0	0	0	0	0	0	0	0	0	0
CO3	3	3	3	2	2	0	0	0	0	0	0	0	0	0
CO4	3	3	3	3	2	0	0	0	0	0	0	0	0	0
CO5	3	3	3	3	3	0	0	0	0	0	0	0	0	0

Course Content

Module 1: Probability and Distributions (12 Lectures)

Random variables, Probability distributions: Discrete & continuous distributions, Binomial, Poisson, Exponential, Normal, Chi-square, t and F distributions. Mathematical expectation and its properties.

Module 2: Descriptive Statistics and Sampling (10 Lectures)

Descriptive measures – central tendency and dispersion, Random sampling, Sampling from finite and infinite populations. Estimates and standard error (with and without replacement). Sampling distribution of sample mean. Stratified random sampling.

Module 3: Linear Statistical Models (7 Lectures)

Scatter diagram. Linear regression and correlation. Least squares method. Rank correlation. Multiple regression & multiple correlation. Analysis of variance – one way, two way (with as well as without interaction).

Module 4: Estimation and Hypothesis Testing (7 Lectures)

Point estimation, Criteria for good estimates (unbiasedness, consistency). Methods of estimation including maximum likelihood estimation. Test of hypothesis – concept and formulation, Type I and Type II errors, Neyman Pearson Lemma, Procedures of testing.

Text Books:

1. **”Probability and Statistics for Engineers”**, R. A. Johnson, Pearson Education.
2. **”Introduction to Mathematical Statistics”**, R. V. Hogg & A. T. Craig, Pearson.

Reference Books:

1. **”Mathematical Statistics”**, J. E. Freund, Pearson.
2. **”Statistics”**, S. P. Gupta, Sultan Chand & Sons.
3. **”Applied Statistics and Probability for Engineers”**, D. C. Montgomery & G. C. Runger, Wiley.
4. **”Fundamentals of Mathematical Statistics”**, S. C. Gupta and V. K. Kapoor, Sultan Chand.

Course Name: Design and Analysis of Algorithms Lab

Course Code: CB391

Contact (Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 1.5

Prerequisites:

1. Data Structures and Algorithms
2. Programming in C/C++

Course Objective(s):

By the end of this course, students will be able to:

- Introduce the fundamental concepts, characteristics, and classifications of algorithms and algorithmic problem solving.
- Develop the ability to analyze and evaluate the efficiency of algorithms using asymptotic notations and recurrence relations.
- Enable students to apply algorithm design strategies such as divide-and-conquer, greedy methods, dynamic programming, and graph algorithms to real-world problems.
- Encourage critical thinking to construct and compare different algorithmic solutions based on performance, correctness, and resource constraints.
- Foster an understanding of computational complexity, NP-completeness, and the implications of intractable problems in computer science and engineering.

Course Outcomes (COs):

After successful completion of the course, students will be able to:

CO1 Explain the foundational concepts of algorithms and illustrate asymptotic analysis using mathematical reasoning to analyze time and space complexity for solving engineering problems.

CO2 Apply recursive and divide-and-conquer techniques to classical problems and analyze their correctness and performance using recurrence relations and empirical methods.

CO3 Design and implement optimized solutions using greedy and dynamic programming approaches and compare alternative algorithms based on computational efficiency and problem constraints.

CO4 Construct solutions for combinatorial problems using backtracking and branch-and-bound, and evalu- ate their effectiveness through modeling and experimentation.

CO5 Interpret the theory of computational complexity classes (P, NP, NP-Complete) and justify the lim- itations of algorithmic solutions with respect to tractability and ethical considerations in technology use.

Course Name: Object-Oriented Programming Using C++ Lab

Course Code: CB392

Contact (Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 1.5

Prerequisites: Basic knowledge of C programming, Understanding of problem-solving techniques and algorithms

Course Objectives: The objectives of this course are to enable students to:

1. To provide practical exposure to object-oriented programming concepts using C++.
2. To develop the ability to implement classes, objects, and advanced OOP features.
3. To strengthen problem-solving and debugging skills through programming exercises.
4. To build small-scale applications involving file handling and error management.

Course Outcomes (COs): After successful completion of the course, students will be able to:

CO1: Apply C++ syntax, structures, and I/O in solving basic programming problems.

CO2: Implement classes, objects, constructors, and operator overloading.

CO3: Develop programs using inheritance, polymorphism, and templates.

CO4: Design programs with exception handling and file operations.

CO5: Integrate multiple OOP concepts into small projects.

CO-PO Mapping:

CO-PO Mapping

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

Course Content / Laboratory Experiments:

1. Programs on basic syntax, data types, operators, and control structures.
2. Implementing classes, objects, constructors, and method overloading.
3. Inheritance: single, multilevel, hierarchical, and use of super.
4. Interfaces and packages.
5. Polymorphism: method overriding and dynamic method dispatch.
6. Exception handling programs.

7. Multithreading concepts with thread creation and synchronization.
8. File handling: text and binary file operations.
9. GUI programming using AWT/Swing.
10. Database connectivity using JDBC.
11. Mini-project on Java application development.

Text/Reference Books:

1. E. Balagurusamy, *Programming with Java*, McGraw Hill Education.
2. Herbert Schildt, *Java: The Complete Reference*, McGraw Hill Education.
3. Cay S. Horstmann, *Core Java Volume I – Fundamentals*, Pearson Education.

Course Name: Database Management Systems Lab

Course Code: CB393

Contact (Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 1.5

Prerequisite:

Basic knowledge of databases and programming fundamentals

Course Objective:

- Understand and practice data definition and data manipulation commands in SQL.
- evelop practical database design using E-R modeling and normalization techniques.
- Implement relational databases using SQL on RDBMS tools.
- Apply PL/SQL programming for procedural constructs.
- Design and implement a mini project applying database concepts.

Course Outcome(s):

After the completion of the course students will be able to

CO1: Construct relational database schema using SQL DDL and DML commands.

CO2: Formulate complex queries using SQL operators, joins, and subqueries.

CO3: Design ER diagrams and normalize database structures.

CO4: Implement procedures, functions, cursors, and triggers using PL/SQL.

CO5: Apply relational database concepts to solve real-world problems.

CO-PO-PSO Mapping

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

Course Content

Module 1: [6L]

Introduction to SQL – DDL, DML, DCL commands; Creating tables; Constraints; Insert, Update, Delete operations.

Module 2: [6L]

Advanced SQL – Aggregate functions, Nested queries, Joins, Set operations, Views.

Module 3: [6L]

Database Design – ER diagram construction and conversion to tables; **Normalization:** 1NF, 2NF, 3NF, BCNF.

Module 4: [6L]

Tuples, Sets, and Dictionaries:
PL/SQL Basics – Variables, Control structures, Cursors, Exception handling.

Module 5: [6L]

PL/SQL Advanced – Procedures, Functions, Triggers.

Module 6: [6L]

Mini Project – Design and implementation of a small-scale DBMS application using SQL and PL/SQL.

Text Books:

1. **”Database System Concepts”**, Abraham Silberschatz, Henry F. Korth, S. Sudarshan, McGraw-Hill.

Reference Books:

1. **”Fundamentals of Database Systems”**, Ramez Elmasri, Shamkant B. Navathe, Pearson.
2. **”An Introduction to Database Systems”**, C. J. Date, Pearson.
3. **”SQL, PL/SQL – The Programming Language of Oracle”**, Ivan Bayross, BPB Publication.

Course Name: Introduction to Java Lab

Course Code: CB392

Contact (Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 1.5

Prerequisites: Basic knowledge of C programming, Understanding of problem-solving techniques and algorithms

Course Objectives: The objectives of this course are to enable students to:

1. To provide practical exposure to object-oriented programming concepts using C++.
2. To develop the ability to implement classes, objects, and advanced OOP features.
3. To strengthen problem-solving and debugging skills through programming exercises.
4. To build small-scale applications involving file handling and error management.

Course Outcomes (COs): After successful completion of the course, students will be able to:

CO1: Implement programs using Java syntax, control structures, and basic OOP principles.

CO2: Apply the concepts of classes, objects, constructors, and method overloading.

CO3: Develop applications using inheritance, interfaces, and polymorphism.

CO4: Apply exception handling, multithreading, and file handling in Java programs.

CO5: Design and implement mini-projects using Java packages, GUI, and database connectivity.

CO-PO Mapping:

CO-PO Mapping

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

Course Content / Laboratory Experiments:

1. Programs on basic syntax, data types, operators, and control structures.
2. Implementing classes, objects, constructors, and method overloading.
3. Inheritance: single, multilevel, hierarchical, and use of super.
4. Interfaces and packages.
5. Polymorphism: method overriding and dynamic method dispatch.
6. Exception handling programs.

7. Multithreading concepts with thread creation and synchronization.
8. File handling: text and binary file operations.
9. GUI programming using AWT/Swing.
10. Database connectivity using JDBC.
11. Mini-project on Java application development.

Text/Reference Books:

1. E. Balagurusamy, *Programming with Java*, McGraw Hill Education.
2. Herbert Schildt, *Java: The Complete Reference*, McGraw Hill Education.
3. Cay S. Horstmann, *Core Java Volume I – Fundamentals*, Pearson Education.

2nd Year 4th Semester									
Sl. No.	Broad Catagory	Catagory	Paper Code	Subject	L	T	P	Total	Credit Points
A. Theory									
1	ENGG	Major	CB401	Software Engineering and Agile Methodologies	3	0	0	3	3
2	ENGG	Major	CB402	Operating Systems	3	0	0	3	3
3	ENGG	Major	CB403	Computer Networks	3	0	0	3	3
4	ENGG	Major	CB404	Formal Language and Automata Theory	3	0	0	3	3
5	ENGG	Minor	BS401	Principles of Management	3	0	0	3	3
6	SCI	Multi	M(CB)401	Linear Algebra	3	0	0	3	3
B. Practical									
1	ENGG	Major	CB491	Software Engineering and Agile Methodologies Lab	0	0	3	3	1.5
2	ENGG	Major	CB492	Operating Systems Lab	0	0	3	3	1.5
3	ENGG	Major	CB493	Computer Networks Lab	0	0	3	3	1.5
4	HUM	Multi	HU(CB)491	Business Communication and Value Science	1	0	2	3	2
C. MANDATORY ACTIVITIES / COURSES									
1	MC	Mandatory Course	MC481	Course from Skill India-I*	0	0	0	0	2 Units
Total of Theory, Practical								30	24.5

Course Name: Software Engineering and Agile Methodologies

Course Code: CB401

Contact: (periods/week): 3

Total Contact Hours: 36

Credits: 3

Prerequisites:

Basic programming knowledge, Understanding of software development life cycle (SDLC) fundamentals.

Course Objective(s):

The objective of the course is to make the students able to –

- Understand software engineering principles, processes, and models.
- Apply structured approaches to software requirement analysis and design.
- Gain knowledge of Agile values, principles, and frameworks.
- Implement Agile practices such as Scrum, Extreme Programming (XP), and Kanban.
- Evaluate the suitability of Agile versus traditional methodologies for different projects.

Course Outcomes (COs):

After successful completion of the course, students will be able to:

CO1: Explain the fundamental concepts of software engineering and process models.

CO2: Apply requirement analysis, system modeling, and design techniques.

CO3: Demonstrate knowledge of Agile principles and practices.

CO4: Apply Scrum, XP, and Kanban frameworks in software development.

CO5: Compare and evaluate Agile and plan-driven methodologies for real-world applications.

CO-PO Mapping

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

Course Content (5 Modules)

Module 1: Introduction to Software Engineering and Process Models (6L)

Nature of software and software engineering Software process and life cycle models (Waterfall, Incremental, Spiral, V-Model) Agile vs. traditional approaches

Module 2: Requirements Engineering (7 Lectures)

Requirement elicitation, analysis, specification, and validation Use case modeling and UML basics Functional vs. non-functional requirements

Module 3: Software Design and Development (7 Lectures)

Principles of good design Design patterns and modularization Coding standards and practices Version control (Git/GitHub basics)

Module 4: Agile Methodologies – Principles and Practices (8 Lectures)

Agile manifesto and principles Scrum: roles, artifacts, ceremonies Extreme Programming (XP): pair programming, TDD, refactoring Kanban and Lean development practices

Module 5: Software Quality and Agile Project Management (8 Lectures)

Software testing: unit testing, integration testing, acceptance testing Continuous integration and delivery (CI/CD) Agile metrics and tools (Jira, Trello) Case studies: Agile vs. traditional approaches in practice

Text Books:

1. **”Software Engineering: A Practitioner’s Approach”**, Roger S. Pressman & Bruce R. Maxim, McGraw Hill, 9th Edition.
2. **”Agile Software Development: Principles, Patterns, and Practices”**, Carlo Ghezzi, Mehdi Jazayeri, Dino Mandrioli.
3. **”Software Engineering: Theory and Practice”**, Robert C. Martin, Pearson Education, 2nd Edition.

Reference Books:

1. **”Agile Estimating and Planning”**, Mike Cohn, Addison-Wesley, 1st Edition.
2. **”Clean Code: A Handbook of Agile Software Craftsmanship”**, Robert C. Martin, Pearson, 1st Edition.
3. **”Software Engineering”**,Ian Sommerville, Pearson Education, 10th Edition.

Course Name: Operating System

Course Code: CB402

Contact(Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 3

Prerequisites: Basic knowledge of computer organization, data structures, and programming in C/C++.

Course Objective(s):

The objective of the course is to make the students able to –

- O1: Understand the fundamental concepts and functions of operating systems.
- O2: Learn process management, CPU scheduling, deadlock handling, and synchronization techniques.
- O3: Study memory management strategies, virtual memory, and file systems.
- O4: Gain knowledge of I/O systems, device management, and protection mechanisms.

Course Outcomes (COs):

After the successful completion of the course, students will be able to:

CO1: Explain the structure, components, and services of modern operating systems.

CO2: Analyze process scheduling, synchronization, and deadlock problems with solutions.

CO3: Apply memory management techniques including paging, segmentation, and virtual memory.

CO4: Demonstrate file system and I/O management concepts.

CO5: Design solutions using OS concepts for resource management, security, and protection.

CO-PO-PSO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	0	0	0	0	0	0	0	0	0	2	0	0
CO2	3	3	2	2	0	0	0	0	0	0	0	2	2	0
CO3	3	2	3	2	2	0	0	0	0	0	0	3	2	2
CO4	2	2	3	3	2	0	0	0	0	0	0	2	3	2
CO5	3	3	3	3	3	0	0	0	0	0	0	3	3	3

Course Contents

Module 1: Introduction to Operating Systems (6 Lectures)

- Functions, types of operating systems
- System calls, operating system structure, OS services

Module 2: Process Management (8 Lectures)

- Process concept, process states, PCB, process scheduling
- Scheduling algorithms: FCFS, SJF, Round Robin, Priority
- Inter-process communication, synchronization (semaphores, monitors)
- Classical synchronization problems, deadlock (detection, prevention, avoidance)

Module 3: Memory Management (8 Lectures)

- Contiguous and non-contiguous memory allocation
- Paging, segmentation, virtual memory, demand paging
- Page replacement algorithms, allocation policies, thrashing

Module 4: File Systems (6 Lectures)

- File attributes, operations, file system implementation
- Directory structure, allocation methods (contiguous, linked, indexed)
- Free space management, protection mechanisms

Module 5: I/O and Secondary Storage Management (8 Lectures)

- I/O hardware, I/O software layers, disk scheduling algorithms
- RAID, device drivers, security and protection issues

NOTE: Assignments & tutorials covering: Scheduling, Deadlock, Memory Management, File Systems, Disk Scheduling.

Text Books:

1. **”Operating System Concepts”**, Abraham Silberschatz, Peter B. Galvin, Greg Gagne, Wiley.
2. **”Operating Systems: Internals and Design Principles”**, William Stallings, Pearson Education.

Reference Books:

1. **”Operating Systems: A Concept-Based Approach”**, D. M. Dhamdhere, McGraw Hill Education.

Course Name: Computer Networks

Course Code: CB403

Contact(Periods/Week): 3 periods

Total Contact Hours: 36

Credit: 3

Course Objective:

- To understand the fundamental principles and modern architectures of computer networks, including design and performance considerations.
- To analyze and compare different types of networks such as Local Area Networks (LANs), Wide Area Networks (WANs), and Wireless LANs (WLANs).
- To apply core networking protocols and standards in solving real-world communication problems.
- To develop basic skills in network programming and socket-based communication.
- To perform and interpret WLAN measurements to evaluate performance and optimize network deployment.

Course Outcomes (COs):

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

CO1: Illustrate the network topologies, model and architecture.

CO2: Apply different networking devices and protocols for problem solving.

CO3: Analyze different networking functions in various layers of OSI and TCP/IP Model.

CO4: Evaluate the optimal route for communication and understand routing algorithms for data transmission.

CO5: Design network architecture and implement it in practical field work.

CO-PO Mapping

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

Course Contents

Module I: Overview of Data Communication and Networking (4L)

- Introduction
- Data communications: components, data representation (ASCII, ISO etc.), direction of data flow (simplex, half duplex, full duplex)
- Network criteria, physical structure (type of connection, topology), categories of network (LAN, MAN, WAN)
- Internet: brief history, Protocols and standards
- Reference models: OSI and TCP/IP

Module II: Physical Layer (5L)

- Overview of data, signal, transmission & transmission media
- Circuit switching: time division & space division switch, TDM bus
- Telephone Network

Module III: Data Link Layer (8L)

- Types of errors, framing, error detection & correction methods
- Flow control
- Protocols: Stop & wait ARQ, Go Back-N ARQ, Selective repeat ARQ, HDLC
- Point to Point Protocol, LCP, NCP, Token Ring
- Reservation, Polling, Multiple access protocols: Pure ALOHA, Slotted ALOHA, CSMA, CSMA/CD, CSMA/CA
- Traditional Ethernet, fast Ethernet

Module IV: Network Layer (7L)

- Internetworking & devices: Repeaters, Hubs, Bridges, Switches, Router, Gateway
- Addressing: IP addressing, subnetting
- Routing: techniques, Routing Protocols, ARP, IP, ICMP, IPV6

Module V: Transport Layer (6L)

- Process to Process delivery
- UDP, TCP
- Congestion Control: Open Loop, Closed Loop choke packets
- Quality of service: techniques to improve QoS – Leaky bucket algorithm, Token bucket algorithm

Module VI: Application Layer (6L)

- DNS, SMTP, SNMP, FTP, HTTPS
- Firewalls, IP Filtering

Text Books:

1. **”Data Communications and Networking” (5th Ed.)**, B. A. Forouzan, TMH.
2. **”Data and Computer Communications” (5th Ed.)**, W. Stallings, PHI/Pearson Education.

Reference Books:

1. **”Computer Networks” (4th Ed.)**, S. Tanenbaum, Pearson Education/PHI.
2. **”Data & Computer Communication”**, Black, PHI.
3. **”Network for Computer Scientists & Engineers”**, Zheng & Akhtar, OUP.

Course Name: Formal Language and Automata Theory

Course Code: CB404

Contact(Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 3

Prerequisite:

Discrete Mathematics, Basic Programming.

Course Objective:

- To introduce the concepts of formal languages and automata theory.
- To understand the power and limitations of computational models.
- To provide a foundation for compiler design and computational complexity.
- To model and analyze computational problems using automata and grammars.

Course Outcome(s):

After the completion of the course students will be able to

CO1: Understand and describe formal languages, automata types, and Chomsky hierarchy.

CO2: Construct finite automata and regular expressions for regular languages.

CO3: Design context-free grammars and pushdown automata for context-free languages.

CO4: Analyze Turing machine models and their computational power.

CO5: Evaluate problems in terms of decidability and computational complexity.

CO-PO-PSO Mapping

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

Course Content

Module 1:

Introduction to Formal Languages and Automata:
Alphabets, Strings, Languages, Grammar types: Type 0 to Type 3 (Chomsky Hierarchy), Finite Automata (FA): Deterministic (DFA) and Non-deterministic (NFA), NFA to DFA conversion, FA with ϵ transitions, Applications of Finite Automata.

Module 2:

Regular Languages and Expressions:
Regular Expressions and their Equivalence with FA, Arden's Theorem, Properties of Regular Languages, Pumping Lemma for Regular Languages, Closure properties and decision problems.

Module 3:

Context-Free Grammars (CFG) and Pushdown Automata (PDA):
Definition and Derivations of CFG, Ambiguity in CFG, Normal Forms (CNF and GNF), PDA: Definition and Types (Deterministic & Non-Deterministic), CFG and PDA equivalence, Applications of PDA.

Module 4:

Turing Machines:
Basic Model of Turing Machine (TM), Language Acceptance by TM, Variants of Turing Machines (Multi-tape, Non-deterministic),Universal Turing Machine, Recursive and Recursively Enumerable Languages.

Text Books:

1. **"Introduction to Automata Theory, Languages, and Computation"**, Hopcroft, Ullman, and Motwani, Pearson Education.
2. **"An Introduction to Formal Languages and Automata"**, Peter Linz, Jones & Bartlett Learning.

Reference Books:

1. **"Formal Languages and Their Relation to Automata"**, John E. Hopcroft and Jeffrey D. Ullman, Addison-Wesley.
2. **"Elements of the Theory of Computation"**, Harry R. Lewis and Christos H. Papadimitriou, Prentice Hall.
3. **"Introduction to Formal Languages, Automata Theory and Computation"**, Kamala Krithivasan and R. Rama, Pearson Education.

Course Name: Principal Of Management

Course Code: BS401

Contact(Periods/Week): 3 periods

Total Contact Hours: 32

Credits: 3

Prerequisite:

NIL

Course Objective(s):

The objective of the course is to make the students able to –

- Understand how managers manage business organizations in a dynamic global environment.
- Explain how organizations develop and maintain competitive advantage.
- Demonstrate how business decisions are made using various tools and techniques to remain competitive.
- Apply problem-solving strategies and critical thinking skills in real-life situations.
- Understand how different areas of business (Manufacturing/Service, Marketing, Finance, and Human Resource Management) support the vision and mission.

Course Outcomes (COs):

After successful completion of the course, students will be able to:

CO1: Explain the circumstances that led to management evolution and their effect on future managers.

CO2: Analyze and evaluate the influence of historical forces on the current practice of management.

CO3: Identify and evaluate social responsibility and ethical issues involved in business situations and logically articulate their position on such issues.

CO4: Develop the process of management's four functions: planning, organizing, leading, and controlling.

CO5: Evaluate leadership styles to anticipate the consequences of each leadership style.

CO-PO Mapping

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

Course Contents

Module 1: Introduction to Management (4L)

- Definition, Nature and Scope, Functions
- Managerial Roles, Levels of Management, Managerial Skills, Challenges of Management
- Evolution of Management: Classical Approach – Scientific and Administrative Management; The Behavioral approach; The Quantitative approach; The Systems Approach; Contingency Approach; IT Approach

Module 2: Planning and Decision Making (10L)

- General Framework for Planning – Planning Process, Types of Plans, Management by Objectives
- Development of Business Strategy
- Decision making and Problem Solving – Programmed and Non-Programmed Decisions, Steps in Prob- lem Solving and Decision Making
- Bounded Rationality and Influences on Decision Making
- Group Problem Solving and Decision Making
- Creativity and Innovation in Managerial Work

Module 3: Leadership, Power and Motivation (10L)

- Leadership, Power and Authority
- Leadership Styles: Behavioral Leadership, Situational Leadership, Leadership Skills
- Motivation – Types of Motivation; Relationship between Motivation, Performance and Engagement
- Content Motivational Theories and Process theories

Module 4: Work Study and Work Measurement (4L)

- Definition of Work Study, Method Study steps, tools and techniques
- Work Measurement: Aims & objectives, stopwatch procedure in Time Study
- Performance rating, allowances and types, calculation of standard time, work sampling

Module 5: Quality Management (6L)

- Definition and importance
- Statistical quality control, acceptance sampling, control charts – mean, range, c, p, np charts
- Zero Defects, Quality Circles, Kaizen, Six Sigma, ISO 9000 implementation steps, Total Quality Management (TQM)

Text Books:

1. **”Essentials of Management”**, Harold Koontz & Heinz Weihrich, McGraw Hill, 10th Edition
2. **”Principles of Management”**, P.C. Tripathi & P.N. Reddy, McGraw Hill Education

Reference Books:

1. **”Management”**, Stephen P. Robbins & Mary Coulter, Pearson Education, Latest Edition
2. **”Principles and Practices of Management”**, L.M. Prasad, Sultan Chand & Sons, Latest Edition
3. **”Management: A Global Perspective”**, Heinz Weihrich & Harold Koontz, McGraw Hill

Course Name: Linear Algebra

Course Code: M(CB)401

Contact(Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 3

Prerequisites

Basic knowledge of algebra and matrix operations.

Course Objective(s):

The objective of the course is to make the students able to –

O1: Understand the theory and computation of matrices, vectors, and their properties.

O2: Solve systems of linear equations using matrix-based methods.

O3: Understand vector spaces, dimension, and orthogonality principles.

O4: Analyze eigenvalues, eigenvectors, and their significance in linear transformations.

O5: Apply concepts like SVD and PCA in practical fields like image processing and machine learning.

Course Outcomes (COs):

After the successful completion of the course, students will be able to:

CO1: Recall fundamental definitions and properties of matrices, determinants, and systems of linear equations.

CO2: Understand concepts of vector spaces, linear dependence, basis, and dimension.

CO3: Apply matrix techniques such as LU decomposition, inverse method, and Cramer’s rule to solve systems of equations.

CO4: Analyze linear transformations and matrix structures using eigenvalues, eigenvectors, and orthogonalization methods like Gram-Schmidt.

CO5: Evaluate the effectiveness of matrix algebra, vector space, singular value decomposition (SVD) and principal component analysis (PCA) in applications like image processing and machine learning.

CO-PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	2	0	0	0	0	0	0	0	0	0
CO2	3	3	2	2	0	0	0	0	0	0	0
CO3	3	3	3	2	2	0	0	0	0	0	0
CO4	3	3	3	3	2	0	0	0	0	0	0
CO5	3	3	3	3	3	0	0	0	0	0	0

Course Contents

Module 1: Matrices and Linear Equations (7 Lectures)

- Introduction to Matrices and Determinants
- Solution of Linear Equations using matrices: Cramer's rule, Inverse of a Matrix, LU decomposition

Module 2: Vector Spaces and Orthogonality (15 Lectures)

- Vector space, linear combinations, span, Dimension
- Basis, Orthogonality, Projections
- Gram-Schmidt orthogonalization, QR decomposition

Module 3: Linear Transformations (9 Lectures)

- Positive definite matrices
- Linear transformations, Matrix representation of linear transformation
- Change of basis

Module 4: SVD and Applications (5 Lectures)

- Singular Value Decomposition (SVD)
- Principal Component Analysis (PCA)
- Applications in Image Processing and Machine Learning

NOTE: Assignments & tutorials covering: Vectors and linear combinations, Matrices, Linear transformations, Solution to $Ax = b$, Determinants, Eigenvalues and Eigenvectors.

Text Books:

1. **”Higher Engineering Mathematics”**, B. S. Grewal, Khanna Publishers.

Reference Books:

1. **”Advanced Engineering Mathematics” (7th Edition)**, Peter V. O'Neil, Cengage Learning.
2. **”Advanced Engineering Mathematics” (2nd Edition)**, Michael D. Greenberg, Pearson.
3. **”Introduction to Linear Algebra” (5th Edition)**, Gilbert Strang, Wellesley-Cambridge Press.
4. **”Applied Mathematics Vol. I & II”**, P. N. Wartikar & J. N. Wartikar, Pune Vidyarthi Griha Prakashan.
5. **”Digital Image Processing”**, R. C. Gonzalez and R. E. Woods, Pearson.
6. Online resource: Introduction to Matrices in Machine Learning

Course Name: Software Engineering Lab

Course Code: CB491

Contact(Periods/Week): 3 periods

Total Contact Hours: 36 Hours

Credits: 1.5

Prerequisites:

Basic knowledge of object-oriented programming and software development lifecycle.

Course Objective(s):

The objective of the course is to make the students able to –

O1: Familiarize with object-oriented technologies and software modeling using UML.

O2: Develop skills in software analysis and design using use cases and class modeling.

O3: Understand and apply various UML diagrams for logical and dynamic modeling.

O4: Bridge the gap between software requirement analysis and object-oriented design.

O5: Gain practical experience with modeling tools and design patterns.

Course Outcomes (COs):

After successful completion of the course, students will be able to:

CO1: Understand object-oriented principles and apply UML to model software systems.

CO2: Analyze system requirements and create Use Case diagrams and actor models.

CO3: Design interaction models using Sequence and Collaboration diagrams.

CO4: Construct static and dynamic UML models such as Class, State, and Activity diagrams.

CO5: Apply design patterns and UML to design distributed and component-based systems.

CO-PO-PSO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	2	2	3	0	0	0	0	0	0	3	2	2
CO2	3	3	3	3	2	0	0	0	0	0	0	3	3	2
CO3	2	3	3	3	3	0	0	0	2	2	0	2	3	3
CO4	2	3	3	3	3	0	0	0	2	2	0	2	2	3
CO5	2	3	3	3	3	0	0	0	2	0	0	3	3	3

Course Content

Module 1: Object-Oriented Concepts & Software Development Models

Object-oriented technologies, UML basics, Waterfall vs Spiral model, Software Crisis and Quality Characteristics, Classes, inheritance, and configurations, Object-Oriented vs Structured Analysis.

Module 2: UML Language and Standards

Elements of UML, Object-Oriented Software Development Process, Overview of UML models, Introduction to Design Patterns, Distributed Systems: Technological aspects.

Module 3: Requirements Analysis Using Use Case Modeling

System requirement analysis, Actor and Use Case definitions, Use Case diagrams and relationships, Writing Use Case goals.

Module 4: Transition from Analysis to Design – Interaction Diagrams

Sequence and Collaboration diagrams, Defining classes, methods, and interfaces, Object identification using Flow of Events.

Module 5: Logical Design – Static Structure Diagrams

Class diagrams: attributes, operations, relationships, Associations, aggregations, generalizations, dependencies, Package diagram and interface modeling.

Module 6: Behavioral Design – Dynamic Models

State and Activity diagrams, Events, transitions, state machines.

Module 7: Component and Deployment Models

Component diagrams: logical and physical views, Deployment diagrams: processors, components, threads, Initial DB design using UML environment.

Text Book:

Object-Oriented Software Engineering: using UML, Patterns, and Java – Bernd Bruegge & Allen H. Dutoit

Reference Book:

Design Patterns: Elements of Reusable Object-Oriented Software – Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides

Course Name: Operating System Lab

Course Code: CB492

Contact(Periods/Week): 3 periods

Total Contact Hours: 36 Hours

Credits: 1.5

Prerequisites:

Knowledge of C programming and basic concepts of operating systems.

Course Objective(s):

- To provide practical exposure to operating system concepts.
- To implement process management, scheduling, and synchronization algorithms.
- To study memory allocation, file systems, and deadlock handling techniques through simulation.
- To enhance programming skills by developing system-level programs.

Course Outcomes (COs):

After successful completion of the course, students will be able to:

CO1: Implement CPU scheduling algorithms using C programs.

CO2: Develop solutions for process synchronization problems.

CO3: Apply deadlock detection and avoidance strategies in simulation programs.

CO4: Demonstrate memory allocation and page replacement algorithms.

CO5: Implement file system operations and develop mini-projects simulating OS functionalities.

CO-PO-PSO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	2	2	0	0	0	0	0	0	0	2	0	0
CO2	3	3	2	2	2	0	0	0	0	0	0	2	2	0
CO3	3	2	3	3	2	0	0	0	0	0	0	3	2	2
CO4	2	2	3	3	3	0	0	0	0	0	0	2	3	2
CO5	3	3	3	3	3	0	0	0	0	0	0	3	3	3

Course Content / Laboratory Experiments

1. Simulation of CPU Scheduling Algorithms: FCFS, SJF, Round Robin, Priority.
2. Implementation of Inter-Process Communication using Shared Memory/Pipes.
3. Implementation of Process Synchronization using Semaphores/Monitors.
4. Simulation of Deadlock Detection and Avoidance Algorithms (Banker’s Algorithm).
5. Implementation of Paging and Segmentation Techniques.
6. Simulation of Page Replacement Algorithms: FIFO, LRU, Optimal.
7. Implementation of File Allocation Techniques: Contiguous, Linked, Indexed.
8. Simulation of Directory Structure and File Operations.
9. Implementation of Disk Scheduling Algorithms: FCFS, SSTF, SCAN, C-SCAN.
10. Mini-project: Design and develop a simulation of an OS component (e.g., Memory Manager, CPU Scheduler, File System).

Text Books:

- Abraham Silberschatz, Peter B. Galvin, Greg Gagne, *Operating System Concepts*, Wiley.
- William Stallings, *Operating Systems: Internals and Design Principles*, Pearson Education.
- D. M. Dhamdhere, *Operating Systems: A Concept-Based Approach*, McGraw Hill Education.

Course Name: Computer Networks Laboratory

Course Code: CB493

Contact(Periods/Week): 3 periods

Total Contact Hours: 36 Hours

Credits: 1.5

Course Objective(s):

O1: Understand the fundamental principles and modern architectures of computer networks, including design and performance considerations.

O2: Analyze and compare different types of networks such as Local Area Networks (LANs), Wide Area Networks (WANs), and Wireless LANs (WLANs).

O3: Apply core networking protocols and standards in solving real-world communication problems.

O4: Develop basic skills in network programming and socket-based communication.

O5: Perform and interpret WLAN measurements to evaluate performance and optimize network deployment.

Course Outcomes (COs):

After successful completion of the course, students will be able to:

CO1: Analyse different network topologies and devices.

CO2: Apply the concept of different type of networking for implementation.

CO3: Implement and discuss the various services offered by transport layer such as TCP and UDP.

CO4: Apply channel allocation, framing, error and flow control techniques.

CO5: Apply the basics of networking protocols for solving real life networking problems.

CO-PO-PSO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	2	2	3	0	0	0	0	0	0	2	0	0
CO2	3	2	3	2	2	0	0	0	0	0	0	2	0	0
CO3	3	3	2	2	3	0	0	0	0	0	0	2	0	0
CO4	3	3	2	3	3	1	1	2	2	0	0	3	0	0
CO5	3	3	3	2	2	1	1	1	1	2	3	3	0	0

Course Contents

1. Familiarization with: Different networking cables, Different connectors, Hubs, Switches, Routers.
2. NIC Installation & Configuration (Windows/Linux).
3. Understanding IP address, subnet etc., Connect the computers in Local Area Network.
4. Study of basic Network Configuration commands.
5. Configure a Network topology using Packet Tracer software.
6. Link Layer Error Detection Mechanism (Cyclic Redundancy Check), Data Link Layer Error Control mechanism (Selective Repeat, Go Back N).
7. Implementation of Data Link Layer Flow Control Mechanism (Stop & Wait, Sliding Window).
8. Server Setup/Configuration: FTP, TELNET, NFS, DNS, Firewall.
9. TCP/UDP Socket Programming: Simple, TCP-based, UDP-based Multicast & Broadcast Sockets.
10. CISCO Packet Tracer Example.

Text Books:

- A. Forouzan – *Data Communications and Networking (5th Ed.)* – TMH.
- W. Stallings – *Data and Computer Communications (5th Ed.)* – PHI/ Pearson Education.

Reference Books:

- A.S. Tanenbaum – *Computer Networks (4th Ed.)* – Pearson Education/PHI.
- Black – *Data & Computer Communication*, PHI.
- Zheng & Akhtar – *Network for Computer Scientists & Engineers*, OUP.

Course Name: Business Communication And Value Science

Course Code: HU(CB)491

Contact(Periods/Week): 2 periods

Total Contact Hours: 48 Hours

Credits: 2

Prerequisites:

Basic knowledge of high school English

Course Objective(s):

Understand the key concepts of business communication.

Understand the key concepts of values and their applications.

Develop effective Listening, Speaking, Reading and Writing skills.

Use the best practices of public speaking in real-life scenarios.

Motivate students to look within and create a better version of self.

Course Outcomes (COs):

After successful completion of the course, students will be able to:

CO1: Recognize the need for business communication.

CO2: Recognize the need for values in profession.

CO3: Apply public speaking in real-life scenarios.

CO4: Understand the basic tenets of communication for personal and professional development.

CO5: Apply the basic communication practices in different types of communication scenarios.

CO-PO-PSO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	1	1	0	0	0	2	3	0	2	1	1	2	0
CO2	3	2	0	0	0	0	0	1	2	3	0	2	0	3
CO3	2	0	0	0	0	0	0	0	3	3	1	0	3	0
CO4	2	0	2	2	3	0	0	0	2	3	2	0	3	0
CO5	2	0	0	0	0	0	2	3	2	3	0	2	0	0

Course Contents

1. **Module 1: Theories of Business Communication and Values in Profession (8L)**
    - Understanding General and Technical Communication
    - Technical writing: Writing flawlessly – understanding common errors
    - Writing Job Application and Cover Letter, CV and Resume
    - Group Discussion: Types and techniques for facing group discussion
    - Values in profession: values and ethics in profession
2. **Module 2: Reading Skills (4L)**
    - Understanding Listening and Reading Skills: Practice of different sub-skills (dictation, listening comprehension, creating podcasts, news reports, pronunciation practice)
3. **Module 3: Speaking Skills (5L)**
    - Understanding speaking etiquettes
    - Practicing JAM sessions, dialogue building, telephonic conversations, situational role plays, skit, debate and group discussion
    - Classroom activity on Book or Movie Review
4. **Module 4: Public Speaking (8L)**
    - Learning public speaking etiquettes and attributes of an effective public speaker
    - Creating a professional presentation
    - Presentation Assignments: general topic, product presentation, idea pitching with poster or other scaffolds (group or individual)
    - Incorporating bar, chart, and diagrams of statistics in presentations
5. **Module 5: Personality Development and Enhancing Constructive Critical Appreciation Skills (5L)**
    - Understanding Self: Personal SWOT analysis
    - Self Introduction
    - Video resume
    - Introduction to PI Skills and Practice Sessions of PI round

Text Books:

- *Technical Communication* by Meenakshi Raman and Sangeeta Sharma; Oxford Publication
- A N Tripathi, *Human Values in the Engineering Profession*, Monograph published by IIM, Calcutta, 1996

Reference Books:

- Mark Ibbotson, *English for Engineering*, Cambridge University Press, 2008
- Fernando A. C, *Business Ethics – An Indian Perspective*, Pearson Publication, 2009
- Prem Vir Kapoor, *Professional Ethics & Human Values*, Khanna Publishing House, New Delhi

3rd Year 5th Semester

Sl. No.	Broad Catagory	Catagory	Paper Code	Subject	L	T	P	Total	Credit Points
A. Theory									
1	ENGG	Major	CB501	Software Design and UML	3	0	0	3	3
2	ENGG	Major	CB502	Machine Learning	3	0	0	3	3
3	ENGG	Major	CB503A	Computer Graphics with Python	3	0	0	3	3
			CB503B	Data Mining and Business Intelligence					
			CB503C	Generative Artificial Intelligence					
			CB503D	Cryptography and Network Security					
3	ENGG	Minor	BA(CB)501A	Financial Computing and Risk Analysis	3	0	0	3	3
			IT(CB)501B	Introduction to Augmented and Virtual Reality					
			IT(CB)501C	E-Commerce and Digital Business Models					
			IT(CB)501D	Soft Computing					
5	HUM	Minor	BS501	Business Accounting and Financial Management	3	0	0	3	3
6	HUM	Minor	BS502	Organizational Behavior	3	0	0	3	3
B. Practical									
1	ENGG	Major	CB591	Software Design and UML Lab	0	0	3	3	1.5
2	ENGG	Major	CB592	Machine Learning Lab	0	0	3	3	1.5
3	ENGG	Major	CB593A	Computer Graphics with Python Lab	3	0	0	3	3
			CB593B	Data Mining and Business Intelligence Lab					
			CB593C	Generative Artificial Intelligence Lab					
			CB593D	Cryptography and Network Security Lab					
4	PRJ	PRJ	CB581	Project-I	0	0	4	4	2
C. MANDATORY ACTIVITIES / COURSES									
1	MC	Mandatory Course	MC581	Course from Skill India-II*	0	0	0	0	2 Units
Total of Theory, Practical								27	24.5

Course Name: Software Design and UML

Course Code: CB501

Contact (Periods/Week): 3:0:0

Total Contact Hours: 36

Credits: 3

Prerequisite:

Basic understanding of object-oriented programming and software engineering concepts.

Course Objective:

The objective of this course is to enable the students to:

- Understand the principles of software design and the role of design in the software development life cycle.
- Learn the fundamentals of object-oriented modeling and design using UML.
- Analyze, design, and model real-world problems using UML diagrams.
- Acquire skills for applying design patterns and best practices in software design.
- Evaluate and compare different design approaches for solving software engineering problems.

Course Outcome(s):

After the completion of the course students will be able to

CO1: Explain software design principles and their significance in software engineering.

CO2: Develop use-case models and identify classes, objects, and relationships using UML.

CO3: Construct structural and behavioral UML diagrams for modeling software systems.

CO4: Apply design patterns and object-oriented principles to solve design problems.

CO5: Evaluate software design alternatives and document the design using UML standards.

CO-PO-PSO Mapping

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

Course Content

Module I: Introduction to Software Design [5L]

Role of software design in software engineering. Principles of good software design (modularity, cohesion, coupling, abstraction). Overview of object-oriented design concepts.

Module II: Introduction to UML [6L]

UML basics and architecture. Building blocks of UML: Things, relationships, and diagrams. Use-case modeling: Actors, use-cases, use-case diagrams.

Module III: Structural Modeling [8L]

Class diagrams: Classes, objects, attributes, methods, relationships. Object diagrams. Packages and component diagrams. Deployment diagrams.

Module IV: Behavioral Modeling [8L]

Interaction diagrams: Sequence and collaboration diagrams. State machine diagrams. Activity diagrams. Communication modeling.

Module V: Advanced UML and Design Patterns [5L]

UML extensibility mechanisms (stereotypes, tagged values, constraints). Introduction to design patterns: Creational, structural, and behavioral patterns. Application of patterns in real-world systems.

Module VI: Case Studies and Best Practices [4L]

Case study of software modeling using UML. Documenting design with UML. Best practices in software design.

Text Books:

1. **”The Unified Modeling Language User Guide”**, Grady Booch, James Rumbaugh, Ivar Jacobson, Pearson.
2. **”Applying UML and Patterns”**, Craig Larman, Pearson.

Reference Books:

1. **”UML Distilled: A Brief Guide to the Standard Object Modeling Language”**, Martin Fowler, Addison Wesley.
2. **”Design Patterns: Elements of Reusable Object-Oriented Software”**, Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides, Addison Wesley.
3. **”Pattern-Oriented Software Architecture”**, Frank Buschmann et al., Wiley.

Course Name: Machine Learning

Course Code: CB502

Contact (Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 3

Prerequisite:

Discrete Mathematics, Probability, Statistics, Linear Algebra, Calculus.

Course Objective:

- Understand fundamental concepts of supervised and unsupervised learning..
- Analyze and implement ML algorithms using real-world datasets.
- Apply machine learning to solve business and industrial problems.
- Evaluate models using appropriate performance metrics.

Course Outcome(s):

After the completion of the course students will be able to

CO1: Understand the theoretical foundations and types of machine learning techniques.

CO2: Apply supervised learning techniques to classification and regression problems.

CO3: Implement unsupervised learning methods for clustering and dimensionality reduction.

CO4: Optimize machine learning models using ensemble methods and hyperparameter tuning.

CO5: Analyze real-world case studies and applications in business and industry domains.

CO-PO-PSO Mapping

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

Course Content

Module 1:

Introduction to Machine Learning:
Definition, types of ML (Supervised, Unsupervised, Reinforcement Learning), applications, pipeline, challenges. Python for ML, NumPy, pandas, scikit-learn.

Module 2:

Supervised Learning:
Linear Regression, Logistic Regression, k-Nearest Neighbors, Decision Trees, Support Vector Machines, Naive Bayes.Evaluation metrics, accuracy, precision, recall, F1-score, ROC-AUC.

Module 3:

Unsupervised Learning:
Clustering; k-Means, Hierarchical, DBSCAN. Dimensionality Reduction; PCA, t-SNE. Anomaly detection, Association rule mining (Apriori, FP-Growth).

Module 4:

Ensemble Methods and Model Optimization:
Bagging, Boosting (AdaBoost, Gradient Boosting, XGBoost), Random Forests. Cross-validation, Grid Search, Bias-Variance tradeoff.

Module 5:

Advanced Topics and Applications: Introduction to Neural Networks and Deep Learning, CNN, RNN (basic). Applications in healthcare, finance, recommendation systems, business analytics.

Text Books:

1. **”Machine Learning”**, Tom M. Mitchell, McGraw-Hill Education.
2. **”Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow”**, Aur´elien G´eron, O’Reilly.

Reference Books:

1.**”Deep Learning”**, Ian Goodfellow, Yoshua Bengio, Aaron Courville, MIT Press.
2. **”Machine Learning: A Probabilistic Perspective”**, Kevin P. Murphy, MIT Press.
3. **”Understanding Machine Learning: From Theory to Algorithms”**, Shai Shalev-Shwartz and Shai Ben-David, Cambridge University Press.

Course Name: Computer Graphics with Python

Course Code: CB503A

Contact (Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 3

Prerequisite:

Basic knowledge of Programming, Understanding of mathematics (geometry, linear algebra, transformations)

Course Objective:

The objective of this course is to enable the students to:

- Understand the fundamentals of computer graphics and its applications.
- Learn 2D and 3D geometric transformations, viewing, and projections.
- Implement computer graphics algorithms using Python libraries.
- Develop interactive graphics programs using OpenGL/Matplotlib/PyOpenGL/Pygame.
- Explore applications of computer graphics in visualization, games, and user interfaces.

Course Outcome(s):

After completion of the course students will be able to

CO1: Explain the principles and pipeline of computer graphics.

CO2: Apply 2D and 3D transformations for geometric modeling.

CO3: Implement graphics algorithms (line, circle, polygon drawing, clipping, shading) in Python.

CO4:Develop programs for viewing, projections, and interactive graphics using Python libraries.

CO5: Design small applications such as games, visualizations, and simulations.

CO-PO-PSO Mapping

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

Course Content

Module 1: Introduction to Computer Graphics (6 Lectures)

Overview and applications of computer graphics Graphics pipeline and display devices Introduction to Python graphics libraries (Matplotlib, Pygame, PyOpenGL)

Module 2: 2D Graphics and Transformations (8 Lectures)

Points, lines, and polygons representation Line and circle drawing algorithms (DDA, Bresenham’s) 2D transformations: translation, rotation, scaling, reflection, shear Homogeneous coordinates and matrix representation.

Module 3: Clipping and 3D Concepts (7 Lectures)

Line clipping (Cohen–Sutherland, Liang–Barsky) Polygon clipping (Sutherland–Hodgman) Introduction to 3D modeling 3D transformations: translation, rotation, scaling.

Module 4: Viewing and Projections (7 Lectures)

3D viewing pipeline Orthographic and perspective projections Hidden surface removal basics Introduction to shading and lighting models.

Module 5: Applications with Python Graphics (8 Lectures)

Visualization using Matplotlib (2D/3D plots) Interactive graphics with Pygame OpenGL programming with PyOpenGL Mini-projects: simple 2D game, 3D visualization, or simulation project.

Text Books:

1. **”Computer Graphics with OpenGL”**, Donald Hearn and M. Pauline Baker, Pearson Education, 4th Edition.
2. **”Interactive Computer Graphics: A Top-Down Approach with WebGL”**, Edward Angel and Dave Shreiner, Addison-Wesley, 7th Edition.
3. **”Fundamentals of Computer Graphics”**, Peter Shirley, CRC Press, 5th Edition.

Reference Books:

1. **”Computer Graphics: Principles and Practice”**, J.D. Foley, A. van Dam, S.K. Feiner, and J.F. Hughes, Addison-Wesley, 3rd Edition.
2. **”Fundamentals of Computer Graphics”**, Peter Shirley, CRC Press, 5th Edition.
3. **”Computer Graphics: A Programming Approach”**, Steven Harrington, McGraw-Hill, 2nd Edition.
4. **”The Scrum Guide”**, Ken Schwaber and Jeff Sutherland, Scrum.org, Latest Edition.
5. **”Learning Python by Building Games with Pygame”**,Alejandro Rodas de Paz & Joseph Howse, Packt, 2nd Edition

Course Name: Data Mining and Business Intelligence

Course Code: CB503B

Contact (Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 3

Prerequisite:

Basic knowledge of statistics, probability, and programming.

Course Objective:

The objective of this course is to enable the students to:

- Introduce the fundamental concepts and modern techniques in data mining.
- Develop an understanding of data preprocessing, transformation, and visualization.
- Apply association, classification, and predictive modeling techniques to business problems.
- Understand the principles and implementation of regression and forecasting models.
- Provide knowledge on prescriptive analytics including optimization and decision support systems.

Course Outcome(s):

After the completion of the course students will be able to

CO1: Understand the principles, techniques, and applications of data mining and business intelligence.

CO2: Preprocess and visualize data effectively using modern tools like Weka.

CO3: Apply association, classification, and prediction algorithms to real-world datasets.

CO4: Design and interpret statistical models and forecasting techniques for business insights.

CO5: Apply optimization and decision analysis models for prescriptive analytics in business scenarios.

CO-PO-PSO Mapping

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

Course Content

Module 1:

Introduction to Data Mining (5L)

- Definition and significance of data mining
- Related technologies – Machine Learning, DBMS, OLAP, Statistics
- Stages of the Data Mining Process
- Data Mining Techniques and Applications
- Knowledge Representation Methods

Module 2:

Data Preprocessing and Representation (7L)

- Data cleaning, transformation, reduction, discretization
- Generating concept hierarchies
- Installing and experimenting with Weka 3
- Knowledge representation and visualization techniques
- Attribute-oriented analysis and statistical measures

Module 3:

Data Mining Algorithms (8L)

- Association rules: itemsets, rule generation, correlation analysis
- Classification: 1R algorithm, Decision Trees, covering rules
- Prediction: Bayesian classification, Bayesian networks, k-NN, Linear Models

Module 4:

Advanced Predictive Analytics (8L)

- Descriptive analytics and trend analysis
- Regression: Simple, Multiple, Logistic, Non-linear, Generalized Linear Models
- Forecasting models: Heuristics, predictive modeling, semiparametric and nonparametric methods

Module 5:

Time Series and Prescriptive Analytics (8L)

- Time Series Analysis: trend, seasonality, smoothing, ARMA/ARIMA models
- Model estimation and forecasting with ARIMA
- Prescriptive Analytics: optimization, network modeling, risk and decision analysis

Home Assignments:

- Experiments with Weka – Visualization, filters, association rules, decision trees, prediction.
- Mining real data – preprocessing and modeling using techniques on domain-specific datasets.
- End-to-end statistical model building – from EDA to model validation and interpretation.
- Build models using linear and nonlinear regression – Logistic, Poisson, Ridge, etc.

Text Books:

1. **”Data Mining: Concepts and Techniques”**, Jiawei Han and Micheline Kamber, Morgan Kaufmann Publishers, 3rd Edition (2010).
2. **”Data Mining and Knowledge Discovery Handbook”**, Lior Rokach and Oded Maimon, Springer, 2nd Edition (2010).
3. **”Time Series Analysis, Forecasting and Control”**, Box, G.E.P. and Jenkins, G.M., Holden-Day (1970).

Reference Books:

1. **”Applied Regression Analysis”**, Draper, N.R. and Smith, H., John Wiley, 3rd Edition (1998).
2. **”Applied Logistic Regression”**, Hosmer, D.W. and Lemeshow, S., Wiley (1989).

Course Name: Generative Artificial Intelligence

Course Code: CB503C

Contact (Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 3

Prerequisite:

Basic understanding of Machine Learning and Deep Learning.

Course Objective:

The objective of this course is to enable the students to:

- Introduce the foundational principles and mathematical underpinnings of Generative AI.
- Explore various generative models including GANs, VAEs, and Transformers.
- Examine the applications of generative AI in text, image, audio, and video synthesis.
- Discuss ethical, legal, and societal implications of generative technologies.
- Familiarize students with industry-standard tools and frameworks for generative modeling.

Course Outcome(s):

After successful completion of the course, students will be able to

CO1: Describe the fundamental concepts of Generative AI and differentiate between GANs, VAEs, and Transformers.

CO2: Analyze the architecture and training mechanisms of deep generative models and evaluate their effectiveness in producing synthetic data.

CO3: Apply generative techniques in domains such as image, text, audio, and video generation, and demonstrate their potential in real-world applications.

CO4: Examine ethical, legal, and societal concerns in generative AI, and justify responsible usage and deployment of generative technologies.

CO5: Utilize tools and frameworks such as TensorFlow, PyTorch, and Hugging Face to design and manage scalable generative AI solutions.

CO-PO-PSO Mapping

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

Course Content

Module 1: Foundations of Generative AI (7L)

Introduction to AI and ML. What is Generative AI? Scope, importance, and trends. Overview of GANs, VAEs, and Transformer-based models. Probability distributions in generative modeling. Common datasets (MNIST, CIFAR-10, COCO). Applications in art, media, healthcare, and marketing.

Module 2: Deep Learning Architectures for Generative Modeling (8L)

Neural network review (FFNNs, CNNs, RNNs). GANs: Generator and discriminator, minimax objective, training instability. VAEs: Latent space, encoder-decoder, KL divergence, reconstruction loss. Transformers: Self-attention, positional encoding, GPT architecture. Applications of GANs/VAEs in structured and unstructured data.

Module 3: Applications of Generative AI (8L)

Text generation: language modeling, completion, summarization (GPT, T5). Image generation: super-resolution, image-to-image translation, style transfer. Audio & music generation: voice cloning, music synthesis. Video generation: deepfakes, automated summarization. Case studies: advertising, entertainment, education.

Module 4: Advanced Techniques and Ethics (7L)

Fine-tuning, transfer learning, domain adaptation. Few-shot and zero-shot learning. Ethical/social concerns: bias, misinformation, societal impact. Legal challenges: IP, copyright, data privacy. AI governance and responsible AI principles.

Module 5: Tools, Frameworks, and Deployment (6L)

Environments: TensorFlow, PyTorch, Colab, Jupyter. Pre-trained models: Hugging Face Transformers. Transfer learning workflows. Deployment: Docker, Kubernetes. Version control, Git, reproducibility, exper- iment tracking.

Text Books:

1. **”Generative Deep Learning”**, David Foster, O’Reilly Media, 2nd Edition.
2. **”Deep Learning”**, Ian Goodfellow, Yoshua Bengio, Aaron Courville, MIT Press.
3. **”Hands-On Generative Adversarial Networks with Keras”**, Rafael Valle, Packt Publishing.

Reference Books:

1. **”Transformers for Natural Language Processing”**, Denis Rothman, Packt.
2. **”Deep Learning with Python”**, Fran¸cois Chollet, Manning Publications, 2nd Edition.
3. **”Machine Learning Engineering”**, Andriy Burkov, True Positive Inc.
4. **”Practical Deep Learning for Cloud, Mobile & Edge”**, Anirudh Koul et al., O’Reilly.
5. **”Building Machine Learning Powered Applications”**, Emmanuel Ameisen, O’Reilly.

Course Name: Cryptography and Network Security

Course Code: CB503D

Contact (Periods/Week): 3:0:0

Total Contact Hours: 36

Credits: 3

Prerequisite:

Basic knowledge of discrete mathematics, probability, and computer networks.

Course Objective:

The objective of this course is to enable the students to:

- Introduce the mathematical foundations of cryptography and their role in secure communication.
- Study classical, symmetric, and asymmetric cryptosystems and their cryptanalysis.
- Understand hash functions, authentication mechanisms, and digital signatures.
- Analyze network security threats and study defense mechanisms and security protocols.
- Explore modern advancements such as Elliptic Curve Cryptography (ECC), Blockchain security, Zero-Trust models, and Post-Quantum Cryptography.

Course Outcome(s):

After successful completion of the course, students will be able to

CO1: Explain mathematical foundations and classical cryptosystems.

CO2: Apply symmetric and asymmetric algorithms for secure communication.

CO3: Analyze vulnerabilities of cryptographic algorithms using cryptanalysis techniques.

CO4: Implement hash functions, MACs, and digital signatures for authentication and integrity.

CO5: Design and evaluate secure network applications using modern cryptographic protocols and agile practices.

CO-PO-PSO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO9	PO10	PSO1	PSO2	PSO3
CO1	3	2	1	1	2	1	1	1	2	3	2
CO2	3	3	2	2	3	2	1	1	3	3	2
CO3	3	3	2	3	3	2	2	2	3	3	2
CO4	2	3	3	2	3	2	3	3	3	3	3
CO5	2	3	3	2	3	3	3	3	3	3	3

Course Content

Module 1: Introduction and Mathematical Foundations (6L)

- Basics of Cryptography and Network Security.
- Number Theory: Modular arithmetic, Euler's theorem, Fermat's theorem.
- Probability, Information Theory, and Randomness in Cryptography.
- Modern Perspective: Entropy measures, computational complexity, randomness testing.

Module 2: Classical Cryptosystems (4L)

- Substitution and Transposition Ciphers.
- Vigen`ere, Playfair, Hill Cipher.
- Cryptanalysis of Classical Ciphers: Frequency analysis, brute-force attacks.

Module 3: Symmetric Key Cryptography (8L)

- Block and Stream Ciphers, Pseudorandom generators.
- Feistel Structure, DES: design, structure, weaknesses.
- AES: Rijndael algorithm, S-boxes, key scheduling.
- Modes of Operation: ECB, CBC, CFB, OFB, CTR.
- Agile Security Perspective: Iterative testing in real-world secure apps.

Module 4: Cryptanalysis of Symmetric Ciphers (4L)

- Differential Cryptanalysis.
- Linear Cryptanalysis.
- Attacks on DES and AES.
- S-box design principles.

Module 5: Hash Functions and MACs (4L)

- Hash Functions: Properties, Merkle–Damgārd construction.
- SHA family (SHA-2, SHA-3).
- MACs: HMAC, CMAC.
- Modern Trends: Password Hashing (PBKDF2, Argon2).

Module 6: Asymmetric Key Cryptography (8L)

- Public Key Principles, Trapdoor Functions.
- RSA: Algorithm, key generation, attacks.
- Discrete Logarithm Problem (DLP), Diffie–Hellman Key Exchange.
- ElGamal Cryptosystem.
- Primality Testing: Fermat, Miller–Rabin.

Module 7: Digital Signatures and Elliptic Curve Cryptography (6L)

- Digital Signatures: RSA, DSS.
- Elliptic Curve Cryptography: Group law, key exchange, signatures.
- Security advantages of ECC.
- Emerging Trend: Post-Quantum Cryptography (lattice-based, code-based).

Module 8: Network Security Protocols and Applications (6L)

- Kerberos, PGP, S/MIME.
- SSL/TLS protocols.
- Malware: Viruses, worms, Trojans.
- Intrusion Detection Systems, Firewalls, VPNs.
- Emerging Security Models: Blockchain-based Security, Zero Trust Architectures.

Laboratory Component

- Implementation of Classical Ciphers (Caesar, Vigen`ere, Hill, Playfair).
- DES and AES implementation.
- RSA key generation, encryption, decryption.
- Diffie–Hellman Key Exchange.
- Implementation of SHA family and HMAC.
- SSL/TLS configuration in a secure communication setup.
- Mini Project: Blockchain-based identity/authentication OR Secure Messaging Application.

Text Books:

1. **”Cryptography and Network Security: Principles and Practice”**, William Stallings, Pearson.
2. **”Introduction to Modern Cryptography”**, Jonathan Katz, Yehuda Lindell, CRC Press.

Reference Books:

1. **”Applied Cryptography”**, Bruce Schneier, Wiley.
2. **”Network Security Essentials”**, William Stallings, Pearson.
3. **”Handbook of Applied Cryptography”**, Alfred Menezes, CRC Press.
4. **”Understanding Cryptography”**, Christof Paar, Jan Pelzl, Springer.

Course Name: Financial Computing and Risk Analysis

Course Code: HU(CB)501A

Contact (Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 3

Prerequisite:

NIL

Course Objective:

The objective of the course is to make the students able to –

- Understand fundamental principles of finance and their application in modern financial systems.
- Evaluate risk-return relationships and time value of money concepts in financial decision-making.
- Comprehend valuation techniques for assets and securities in various market conditions.
- Develop critical thinking skills for financial problem-solving in technology-driven environments.

Course Outcome(s):

After successful completion of the course, students will be able to

CO1: Apply core financial theories and principles to analyse financial markets and institutions.

CO2: Evaluate investment opportunities using appropriate valuation methods and risk assessment techniques.

CO3: Understand financial intermediation and the role of various market participants.

CO4: Interpret financial statements and use financial ratios for decision-making.

CO5: Assess the impact of technology on financial systems, markets, and services.

CO-PO-PSO Mapping

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

Course Content

Module 1:

Introduction (4L)
Introduction to financial markets, financial instruments, bonds, stocks, and financial derivatives.

Module 2:

Time Value of Money (6L)
Simple and compound interest rate, Net Present Value, Internal Rate of Return, and annuities.

Module 3:

Foundations of Financial Risk Management (10L)
Introduction to financial risk management; Risk management frameworks and governance; Risk identification and classification; Risk appetite and tolerance; Enterprise risk management (ERM) frameworks; Risk culture and organizational aspects.

Module 4:

Risk Analysis (16L)
Market Risk Measurement and Management: Sources and types of market risk, Expected Shortfall and other coherent risk measures, Scenario analysis and stress testing, Market risk mitigation strategies and hedging.
Credit Risk Analysis and Management: Credit risk components and exposures, Credit scoring and rating methodologies, Credit portfolio management.
Operational and Liquidity Risk Management: Operational risk identification and assessment, Operational risk quantification methods, Liquidity risk measurement.

Text Books:

1. **”Risk Management and Financial Institutions”** – J.C. Hull, 6th Edition, Wiley Finance, 2023.
2. **”Quantitative Risk Management: Concepts, Techniques and Tools”** – A.J. McNeil, R. Frey, P. Embrechts, 2nd Edition, Princeton University Press, 2023.

Reference Books:

1. **”Central Counterparties: Mandatory Clearing and Bilateral Margin Requirements for OTC Derivatives”** – J. Gregory, 2nd Edition, Wiley Finance, 2023.
2. **”Elements of Financial Risk Management”** – P.F. Christoffersen, 3rd Edition, Academic Press, 2023.
3. **”Investments”** – Z. Bodie, A. Kane, and A. Marcus, 10th Edition, McGraw-Hill, 2013.

Course Name: E-Commerce And Digital Business Models

Course Code: IT(CB)501C

Contact (Periods/Week): 3 periods

Total Contact Hours: 37

Credits: 3

Prerequisite:

Basic understanding of business processes and web technologies.

Course Objective:

The objective of this course is to enable the students to:

- Understand the fundamental concepts of e-Commerce, e-Business, and digital business models.
- Explore e-marketplaces, auction models, and their economic and competitive impact.
- Examine e-business applications like ERP, e-SCM, CRM, and digital payment systems.
- Analyze the influence of e-commerce across industries such as tourism, finance, education, and publishing.
- Introduce concepts of e-Government and launching online business initiatives including SEO and web design.

Course Outcome(s):

After the completion of the course students will be able to

CO1: Define e-Commerce and e-Business and describe various digital business models.

CO2: Analyze the structure, mechanisms, and types of e-marketplaces and auctions.

CO3: Apply knowledge of ERP, CRM, e-Procurement, and payment systems in digital business contexts.

CO4: Evaluate the impact of e-commerce in various industries like tourism, finance, and education.

CO5: Design and evaluate online business models, websites, and SEO strategies.

CO-PO-PSO Mapping

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

Course Content

Module 1:

Introduction to E-Business and E-Commerce (7L, 15%)

1.01 Definitions: e-Commerce, e-Business, types of EC transactions

1.02 E-Business Models and Elements

1.03 Internet Marketing and e-Tailing

1.04 Benefits and limitations of e-Commerce

Module 2:

E-Marketplaces – Structures, Mechanisms, Economics & Impacts (6L, 15%)

2.01 Functions and features of e-Marketplaces

2.02 Types of auctions and their characteristics

2.03 Wireless e-commerce and its impact

2.04 Competition in the digital economy

Module 3:

E-Business Applications, E-Procurement & E-Payment Systems (7L, 15%)

3.01 ERP, e-SCM, CRM, and integration tools

3.02 E-Procurement: definitions, methods, benefits

3.03 Smart cards and B2B payment systems

Module 4:

Impact of E-Business Across Industries (7L, 10%)

4.01 Online applications: Tourism, Real Estate, Publishing, e-Grocers, Banking

4.02 Digital product delivery and On-demand systems

Module 5:

E-Learning and Online Education (5L, 18%)

5.01 E-Learning definition, industry and benefits

5.02 Content development tools and technologies

5.03 Delivery approaches and evaluation methods

Module 6:

E-Government (4L, 12%)

6.01 Definition, services, implementation

6.02 Challenges and case studies

Module 7:

Launching Online Business & E-Commerce Projects (6L, 10%)

7.01 Requirements and funding for online business

7.02 Managing website development

7.03 Search engine optimization and web evaluation

Module 8:

Mobile Commerce (M-Commerce) (3L, 5%)

8.01 Introduction to mobile commerce

8.02 Applications and business trends

Text Book:

1. **”Electronic Commerce”**, Gary Schneider, Cengage Learning.

Reference Books:

1. **”Digital Business and E-Commerce Management”**, Dave Chaffey, Pearson.
2. **”E-Business and E-Commerce Management”**, Henry Chan, Wiley.
3. **”Introduction to E-Commerce and Digital Business Models”**, (Typically from Turban, King, or Laudon & Traver).
4. Relevant case studies from Harvard Business Review on Digital Business.

Course Name: Soft Computing

Course Code: IT(CB)501D

Contact (Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 3

Prerequisite:

Concept of programing language, File handeling, Mathametics.

Course Objectives

- To understand the concepts of soft computing techniques: fuzzy logic, neural networks, and genetic algorithms.
- To apply soft computing approaches in real-world problems.
- To compare and analyze different soft computing strategies for a given problem.

Course Outcomes (COs)

After the completion of the course students will be able to

- CO1: Understand the basics of soft computing, its components, and applications.
- CO2: Design and implement fuzzy logic systems for reasoning under uncertainty.
- CO3 : Construct and train artificial neural networks for pattern recognition tasks.
- CO4 : Apply genetic algorithms to solve optimization problems.
- CO5 : Integrate various soft computing tools for hybrid intelligent systems.

CO–PO Mapping

CO-PO-PSO Mapping

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

Syllabus Content

Module 1: Introduction to Soft Computing

- Concept of computing systems,
- Hard vs. Soft computing,
- Soft computing techniques and applications,
- Need and relevance in real-world scenarios

Module 2: Fuzzy Logic Systems

- Fuzzy sets and membership functions,
- Fuzzy logic operators and rules,
- Fuzzy inference systems,
- Mamdani and Sugeno models,
- Applications in control and decision-making.

Module 3: Artificial Neural Networks (ANN)

- Biological neuron model
- Perceptron, Multilayer perceptron (MLP),
- Backpropagation algorithm, Self-Organizing Maps (SOM)
- Applications in classification and pattern recognition

Module 4: Genetic Algorithms (GA)

- Evolutionary computing concepts
- Chromosomes, selection, crossover, mutation
- Fitness functions and population evolution
- Applications in optimization

Module 5: Hybrid Systems and Applications

- Integration of ANN, Fuzzy, and GA
- Neuro-Fuzzy systems
- Real-life case studies in prediction, optimization, and control systems

Textbooks

1. S. N. Sivanandam and S. N. Deepa, *Principles of Soft Computing*, Wiley India.
2. J.S.R. Jang, C.T. Sun, and E. Mizutani, *Neuro-Fuzzy and Soft Computing*, Pearson Education.

Reference Books

1. Simon Haykin, *Neural Networks and Learning Machines*, Pearson.
2. David E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*, Pearson.
3. Timothy J. Ross, *Fuzzy Logic with Engineering Applications*, Wiley.

Course Name: Business Accounting and Financial Management

Course Code: BS501

Contact (Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 3

Prerequisite:

NIL

Course Objective:

The objective of the course is to make the students able to –

- Understand fundamental concepts of accounting and financial statements.
- Learn to apply accounting principles in business decision-making.
- Analyze the cost structure and prepare budgets.
- Interpret financial statements using ratio and cash flow analysis.
- Apply principles of capital budgeting and financial planning.

Course Outcome(s):

After successful completion of the course, students will be able to

CO1: Explain core concepts of accounting and double-entry bookkeeping.

CO2: Prepare financial statements and perform journal & ledger posting.

CO3: Analyze cost structures and prepare various budgets.

CO4: Apply financial analysis tools to interpret financial health.

CO5: Make informed decisions using capital budgeting and working capital concepts.

CO-PO-PSO Mapping

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

Course Content

Module 1:

Basics of Accounting (8L)
Principles, types of accounts, Journal, ledger, Trial balance, Final Accounts.

Module 2:

Cost Accounting (8L)
Cost concepts, Cost sheet, Marginal costing, Contract costing.

Module 3:

Financial Analysis (6L)
Ratio analysis, common-size statements, comparative analysis.

Module 4:

Budgeting & Planning (6L)
Cash, flexible, production and sales budgets.

Module 5:

Capital Budgeting and Investment Analysis (8L)
Time Value of Money, Traditional & Modern methods of Investment analysis.

Text Books:

1. **”Financial Management”** – I.M. Pandey, Vikas Publishing, Latest edition.
2. **”Management Accounting”** – S.N. Maheshwari, Sultan Chand, Latest edition.

Reference Books:

1. **”Principles of Financial Management”** – Prasanna Chandra, McGraw-Hill, Latest edition.
2. **”Financial Accounting”** – Jain & Narang, Kalyani Publishers, Latest edition.
3. **”Accounting for Managers”** – T.P. Ghosh, Taxman, Latest edition.

Course Name: Organizational Behavior

Course Code: BS502

Contact (Periods/Week): 3 periods

Total Contact Hours: 34

Credits: 3

Prerequisite:

None

Course Objective:

The objective of this course is to make the students able to:

- Understand the fundamental concepts, models, and processes of Organizational Behaviour in modern organizations.
- Analyze the influence of perception, personality, motivation, and attitudes on individual and group behaviour.
- Explore group dynamics, power structures, leadership, and political behaviour in organizational settings.
- Apply behavioural models and techniques to design high-performance work systems.
- Integrate cross-cultural, ethical, and technological dimensions into decision-making related to human behaviour in organizations.

Course Outcome(s):

After successful completion of the course, students will be able to

CO1: Explain key concepts and environmental influences on Organizational Behaviour including culture, globalization, and ethics.

CO2: Analyze individual-level behavioral processes such as perception, attribution, personality, and emotional intelligence.

CO3: Evaluate group behaviour, leadership dynamics, and the role of power and politics in organizational effectiveness.

CO4: Apply motivational theories and behavioral management tools for designing high-performance teams and workplace practices.

CO5: Demonstrate the ability to integrate the principles of organizational behaviour for effective management, change, innovation, and engineering practice.

CO-PO-PSO Mapping

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

Course Content

Module 1: Introduction to Organizational Behaviour and Cognitive Processes (8L)

Definition, nature, and scope of OB. Environmental and organizational context of OB. Impact of IT, globalization, diversity, ethics, culture, reward systems, and organizational design. Cognitive processes: perception and attribution, perceptual selectivity and organization, social perception, attribution theories (Heider, Kelley), locus of control, attribution errors, impression management.

Module 2: Personality and Attitudes (10L)

Personality as a continuum, Johari Window, Transactional Analysis. Attitudes – nature and dimensions, job satisfaction, organizational commitment. Motivational needs and processes, theories of motivation (content and process), motivation across cultures. Positive organizational behaviour: optimism, emotional intelli- gence, self-efficacy.

Module 3: Dynamics of OB (6L)

Power and politics: meaning and types of power, empowerment. Groups vs teams – nature, dynamics of informal groups, dysfunctions of groups/teams, modern workplace team dynamics.

Module 4: Leading High Performance (4L)

Job design and goal setting for high performance. Socio-technical design, high-performance work practices. Behavioural performance management: reinforcement, punishment, behavioural modification. Leadership theories, styles, skills of effective leaders.

Module 5: Organization Development and Culture (6L)

Forces for change, resistance to change, planned change models. Work culture and climate, intrinsic and extrinsic factors for quality of work life. Case study.

Text Books:

1. **”Organizational Behaviour”**, Stephen P. Robbins & Timothy A. Judge, Pearson.
2. **”Organizational Behaviour”**, Uma Sekaran, Tata McGraw Hill.

Reference Books:

1. **”Human Behaviour at Work: Organizational Behaviour”**, Keith Davis.
2. **”Understanding Organizational Behaviour”**, Udai Pareek, Oxford University Press.
3. **”Organizational Behaviour”**, McShane and Von Glinow, Tata McGraw Hill.

Course Name: Software Design and UML Laboratory

Course Code: CB591

Contact (Periods/Week): 0:0:3

Total Contact Hours: 36

Credits: 1.5

Prerequisite:

Basic knowledge of object-oriented programming and Software Design (CB501).

Course Objective:

The objective of this course is to enable the students to:

- Understand the concepts of software design and modeling using UML.
- Develop hands-on skills in preparing UML diagrams for real-world problems.
- Apply object-oriented design principles to represent systems effectively.
- Analyze requirements and convert them into suitable design models.
- Gain exposure to software design tools for UML modeling and documentation.

Course Outcome(s):

After the completion of the course students will be able to

CO1: Create UML diagrams such as use case, class, sequence, and activity diagrams.

CO2: Apply object-oriented principles in modeling system architecture.

CO3: Model dynamic and interactive system behavior using sequence and state diagrams.

CO4: Use UML tools to document and validate design artifacts.

CO5: Demonstrate design skills by developing mini-projects using UML modeling.

CO-PO-PSO Mapping

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

Course Content

1. Introduction to UML and installation of UML modelling tools.
2. Developing Use Case Diagrams for a sample problem statement.
3. Creating Class Diagrams with attributes, methods, and relationships.
4. Drawing Object Diagrams for static snapshots of systems.
5. Preparing Sequence Diagrams to represent interactions.
6. Creating Collaboration/Communication Diagrams.
7. Modelling system behavior using State Machine Diagrams.
8. Developing Activity Diagrams to represent workflows.
9. Designing Component and Deployment Diagrams.
10. Mini-project: Model a real-world application (e.g., Library System, Online Shopping, Banking System) with a full set of UML diagrams.

Text Books:

1. **”The Unified Modeling Language User Guide”**, Grady Booch, James Rumbaugh, Ivar Jacobson, Pearson.
2. **”Applying UML and Patterns”**, Craig Larman, Pearson.

Reference Books:

1. **”UML Distilled: A Brief Guide to the Standard Object Modeling Language”**, Martin Fowler, Addison Wesley.
2. **”Design Patterns: Elements of Reusable Object-Oriented Software”**, Erich Gamma, Richard Helm, Ralph Johnson, John Vlissides, Addison Wesley.
3. **”Pattern-Oriented Software Architecture”**, Frank Buschmann et al., Wiley.

Course Name: Machine Learning Lab

Course Code: CB592

Contact (Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 1.5

Prerequisite:

Concept of programing language, File handeling, Mathametics.

Course Objective:

- To implement and understand the functioning of core machine learning algorithms.
- To gain hands-on experience using Python and ML libraries (e.g., Scikit-learn).
- To evaluate performance metrics and improve model accuracy.
- To build and test models using real-world datasets.

Course Outcome(s):

After the completion of the course students will be able to

CO1: Use Python libraries to perform data manipulation and visualization.

CO2: Implement various supervised learning algorithms and assess their performance.

CO3: Apply unsupervised learning methods and interpret clustering results.

CO4: Design, implement, and evaluate machine learning workflows on real-world datasets.

CO5: Demonstrate teamwork and presentation skills through a mini-project.

CO-PO-PSO Mapping

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

Course Content

1. Introduction to Python libraries for ML: NumPy, pandas, matplotlib, seaborn.
2. Data preprocessing: Handling missing data, encoding, normalization, train-test split.
3. Implement Linear Regression and evaluate performance (MSE, R^2).
4. Implement Logistic Regression for binary classification.
5. Implement k-Nearest Neighbors and tune k using cross-validation.
6. Train and visualize Decision Tree and Random Forest classifiers.
7. Perform classification using Support Vector Machines (SVM).
8. Perform Unsupervised Learning using k-Means and plot clusters.
9. Apply PCA for dimensionality reduction and visualize results.
10. Implement ensemble learning techniques: Bagging and Boosting.
11. Build and evaluate a simple Artificial Neural Network using Keras/TensorFlow.
12. Mini Project: Apply ML pipeline on a real-world dataset (e.g., loan prediction, sentiment analysis, etc.)

Text Books:

1. **”Machine Learning”**, Tom M. Mitchell, McGraw-Hill Education.
2. **”Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow”**, Aur´elien G´eron, O’Reilly.

Reference Books:

1. **”Deep Learning”**, Ian Goodfellow, Yoshua Bengio, Aaron Courville, MIT Press.
2. **”Machine Learning: A Probabilistic Perspective”**, Kevin P. Murphy, MIT Press.
3. **”Understanding Machine Learning: From Theory to Algorithms”**, Shai Shalev-Shwartz and Shai Ben-David, Cambridge University Press.

Course Name: Computer Graphics with Python Lab

Course Code: CB593A

Contact (Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 1.5

Prerequisite:

- Basic programming in C/C++ or Python
- Understanding of matrices and transformations
- Familiarity with IDEs, basic debugging, and Git version control

Course Objective:

The objective of this course is to enable the students to:

- Provide hands-on experience with fundamental drawing and transformation algorithms in computer graphics.
- Develop the ability to simulate 2D/3D object behavior through programming.
- Integrate agile practices like backlog creation, sprint planning, and iteration in software development.
- Foster team-based project development following the Agile Scrum methodology.
- Enable the design and implementation of mini-projects simulating real-world graphical applications.

Course Outcome(s):

After completion of the course students will be able to:

CO1: Implement basic line, circle, and ellipse drawing algorithms using raster techniques.

CO2: Develop programs to perform 2D and 3D geometric transformations.

CO3: Simulate viewing, clipping, and hidden surface removal techniques.

CO4: Construct simple rendering and shading models in software.

CO5: Apply agile principles to manage mini graphics projects through iterative sprints.

CO-PO-PSO Mapping

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

Course Content (List of Suggested Experiments)

- **Lab 1: Introduction to Graphics with Python** Setting up Python environment for graphics (Matplotlib, Pygame, PyOpenGL) Drawing basic shapes: points, lines, circles using Python libraries
- **Lab 2: Line Drawing Algorithms** Implementation of DDA and Bresenham's line drawing algorithm in Python Comparison of accuracy and efficiency
- **Lab 3: Circle and Polygon Drawing** Bresenham's circle drawing algorithm Polygon filling techniques (scan-line method)
- **Lab 4: 2D Transformations – I** Translation and scaling of 2D objects Matrix representation of transformations
- **Lab 5: 2D Transformations – II** Rotation, reflection, and shear transformations Combined transformations and homogeneous coordinates
- **Lab 6: Line and Polygon Clipping** Cohen–Sutherland line clipping algorithm Liang–Barsky line clipping algorithm Sutherland–Hodgman polygon clipping
- **Lab 7: 3D Transformations – I** 3D translation, scaling, and rotation using Python Matrix operations for 3D transformations
- **Lab 8: 3D Viewing and Projections** Orthographic projection implementation Perspective projection with simple 3D objects
- **Lab 9: Visualization using Matplotlib (2D/3D)** Plotting 2D and 3D data Surface plots, wireframes, contour plots
- **Lab 10: Interactive Graphics with Pygame** Event handling and animation basics Creating a simple 2D game using sprites
- **Lab 11: Introduction to PyOpenGL** Basics of OpenGL with Python Rendering simple 3D shapes with transformations

Text Books:

1. **”Computer Graphics with OpenGL”**, Donald Hearn and M. Pauline Baker, Pearson Education, 4th Edition.
2. **”Interactive Computer Graphics: A Top-Down Approach with WebGL”**, Edward Angel and Dave Shreiner, Addison-Wesley, 7th Edition.
3. **”Fundamentals of Computer Graphics”**, Peter Shirley, CRC Press, 5th Edition.

Reference Books:

1. **”Computer Graphics: Principles and Practice”**, J.D. Foley, A. van Dam, S.K. Feiner, and J.F. Hughes, Addison-Wesley, 3rd Edition.
2. **”Computer Graphics: A Programming Approach”**, Steven Harrington, McGraw-Hill, 2nd Edition.
3. **”Computer Graphics Using OpenGL”**, F.S. Hill Jr. and Stephen M. Kelley, Pearson Education, 3rd Edition.
4. **”The Scrum Guide”**, Ken Schwaber and Jeff Sutherland, Scrum.org, Latest Edition.
5. **”User Stories Applied: For Agile Software Development”**, Mike Cohn, Addison-Wesley, 1st Edition.

Course Name: Data Mining And Business Intelligence Lab

Course Code: CB593B

Contact (Periods/Week): 0:0:3

Total Contact Hours: 36

Credits: 1.5

Prerequisite:

Basic understanding of algorithms, statistics, and programming.

Course Objective:

The objective of this course is to enable the students to:

- Understand the concepts and process of Knowledge Discovery in Databases (KDD).
- Implement data mining algorithms like Apriori, Decision Tree, and OneR.
- Gain hands-on experience using tools such as Weka, XLMiner, and Hadoop.
- Apply classification, association, and prediction techniques on real-world datasets.
- Evaluate model accuracy, bias, and complexity through pruning and cost-sensitive learning.

Course Outcome(s):

After the completion of the course students will be able to

CO1: Demonstrate understanding of the KDD process and data mining techniques.

CO2: Apply association rule mining and classification algorithms using tools like Weka.

CO3: Analyze real datasets using Decision Trees, rule-based classifiers, and OneR.

CO4: Evaluate model performance using cross-validation, pruning, and cost-sensitive learning.

CO5: Compare and interpret different models with a focus on simplicity, bias, and error trade-offs.

CO-PO-PSO Mapping

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

Course Content

Lab Topics

- Introduction to the Knowledge Discovery in Databases (KDD) Process
- Implementation of the Apriori Algorithm in C
- Association Rule Mining using contactlenses.arff in Weka
- Classification using student.arff and J48 Decision Tree
- Association Rules using supermarket.arff in Weka
- Study of data mining tools: Weka, XLMiner, and Hadoop
- Introduction to Mining Techniques: Classification, Association, Prediction
- Decision Trees and Neural Networks – Concepts and Application

List of Experiments

1. Identify and separate categorical vs. real-valued attributes in a dataset
2. Attribute selection and rule formulation for credit assessment
3. Train a Decision Tree and report the generated model
4. Test the model on the training dataset and compute accuracy
5. Discuss limitations of testing on training data
6. Apply cross-validation and compare performance
7. Attribute bias analysis – exclude personal-status and foreign-worker attributes
8. Reduce dimensionality and test classification using key attributes
9. Cost-sensitive learning using cost matrix in Weka for misclassification penalties
10. Apply Reduced Error Pruning and evaluate pruned models
11. *(Extra Credit)* Convert Decision Tree to if-then-else rules; compare J48, PART, and OneR classifiers

Text Book:

1. **”Data Mining: Concepts and Techniques”**, Han, Jiawei & Kamber, Micheline, Morgan Kaufmann Publishers, 3rd Edition (2010).

Reference Books:

1. **”Data Mining and Knowledge Discovery Handbook”**, Lior Rokach & Oded Maimon, Springer, 2nd Edition (2010).
2. **”Python Data Analysis”**, Armando Fandango, Packt Publishing.
3. **”Data Analysis with Python: A Modern Approach”**, David Taieb, Packt Publishing.

Course Name: Generative Artificial Intelligence Lab

Course Code: CB593C

Contact (Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 1.5

Prerequisite:

Basic knowledge of Python, Machine Learning fundamentals.

Course Objective:

The objective of this course is to enable the students to:

- Introduce foundational and advanced hands-on techniques in generative AI using modern tools and frameworks.
- Build practical skills in implementing GANs, VAEs, and transformer models.
- Expose students to real-world applications such as image, text, audio, and video generation.
- Ensure students can ethically evaluate and deploy generative models.
- Prepare students for industry-based problem solving using generative AI.

Course Outcome(s):

After the completion of the course students will be able to

CO1: Implement and train generative AI models like GANs, VAEs, and Transformers.

CO2: Apply generative techniques to real-world problems (text, image, and sound).

CO3: Demonstrate ethical reasoning in generative model development.

CO4: Collaborate effectively in teams to design, develop, and deploy AI systems.

CO5: Showcase industry-readiness through final project presentation and documentation.

CO-PO-PSO Mapping

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

Course Content

Lab No.	Title	Tools/Frameworks
1	Introduction to Generative Models: Dataset visualization (MNIST, CIFAR-10)	Python, matplotlib, seaborn
2	Implementing a Simple GAN for MNIST Digit Generation	TensorFlow/PyTorch
3	Visualizing Generator and Discriminator Training Dynamics	TensorBoard/matplotlib
4	Building and Training a Variational Autoencoder (VAE)	PyTorch
5	Image Style Transfer using Convolutional Neural Networks	OpenCV, PyTorch
6	Text Generation using GPT-style Transformers	Hugging Face Transformers
7	Music/Audio Generation using Pretrained Models	Magenta, torchaudio
8	Super-Resolution Image Generation using ES-RGAN	ESRGAN, PyTorch
9	Few-shot Learning with Pretrained Models (e.g., GPT, T5)	Hugging Face, TensorFlow
10	Ethical Considerations in Generative AI: Bias Testing in Text/Image Generation	Custom dataset & prompt engineering
11	Deployment of Generative Models using Docker & Streamlit	Docker, Streamlit
12	Mini Capstone Lab: Creative Project Showcase (team-based generative solution)	Any applicable tools

Text Books:

1. **”Deep Learning with Python”**, Fran¸cois Chollet, Manning Publications, 2nd Edition.
2. **”Generative Deep Learning”**, David Foster, O’Reilly Media, 2nd Edition.
3. **”Hands-On Generative Adversarial Networks with Keras”**, Rafael Valle, Packt Publishing.

Reference Books:

1. **”Deep Learning”**, Ian Goodfellow, Yoshua Bengio, Aaron Courville, MIT Press.
2. **”Practical Deep Learning for Cloud, Mobile & Edge”**, Anirudh Koul et al., O’Reilly.
3. **”Transformers for Natural Language Processing”**, Denis Rothman, Packt.
4. **”Machine Learning Engineering”**, Andriy Burkov, True Positive Inc.
5. **”Building Machine Learning Powered Applications”**, Emmanuel Ameisen, O’Reilly.

Course Name: Cryptography and Network Security Lab

Course Code: CB593D

Contact (Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 1.5

Prerequisite:

Basic Programming, Computer Networks.

Course Objectives

- Understand various cryptographic algorithms and their applications.
- Implement encryption and decryption using classical and modern techniques.
- Gain practical exposure to network security tools and protocols.
- Analyze and simulate network attacks and secure countermeasures.

Course Outcomes (COs)

After the completion of the course students will be able to

CO1: Apply classical encryption techniques to encrypt/decrypt data.

CO2: Implement modern cryptographic algorithms (DES, AES, RSA, etc.).

CO3 : Analyze key exchange and hashing algorithms.

CO4 : Simulate and analyze various types of cyber attacks and their countermeasures.

CO5 : Use network security tools like Wireshark, Snort, VPN for practical applications.

CO–PO Mapping

CO-PO-PSO Mapping

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

List of Experiments (Indicative)

1. Implementation of Caesar Cipher, Monoalphabetic and Playfair Cipher.
2. Implementation of Hill Cipher and Vigen`ere Cipher.
3. Implementation of DES (Data Encryption Standard) algorithm.
4. Implementation of AES (Advanced Encryption Standard) algorithm.
5. Implementation of RSA algorithm for secure data transmission.
6. Implementation of Diffie-Hellman Key Exchange.
7. Hashing techniques using SHA-1, SHA-256, and MD5.
8. Implement digital signatures using DSA or RSA.
9. Packet Sniffing using Wireshark – Analysis of TCP, UDP, and ICMP packets.
10. Implementation of IPsec and SSL/TLS concepts using open tools.
11. Demonstrate SQL Injection attack and prevention techniques.
12. Demonstration of firewall and IDS using tools like Snort.
13. Configure VPNs using open-source software (e.g., OpenVPN).
14. Simulate Man-in-the-Middle (MITM) attack and mitigation.

Textbooks

1. William Stallings, *Cryptography and Network Security – Principles and Practices*, Pearson Education.
2. Behrouz A. Forouzan, *Cryptography and Network Security*, McGraw Hill.

Reference Books

1. Charlie Kaufman, Radia Perlman, *Network Security: Private Communication in a Public World*, Pearson.
2. Atul Kahate, *Cryptography and Network Security*, Tata McGraw Hill.
3. Bruce Schneier, *Applied Cryptography: Protocols, Algorithms, and Source Code in C*, Wiley.

Course Name: Computational Statistics

Course Code: M(CB)601

Contact (Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 3

Prerequisite:

Basic knowledge of probability, linear algebra, and statistical inference.

Course Objective:

The objective of the course is to make the students able to –

- Understand the theory and properties of multivariate normal distributions.
- Apply regression analysis in multivariate contexts and validate model assumptions.
- Analyze data using principal component and factor analysis techniques.
- Evaluate classification performance using discriminant analysis methods.
- Implement clustering algorithms and interpret cluster-based segmentation results.

Course Outcome(s):

After successful completion of the course, students will be able to:

CO1: Remember key concepts and properties of multivariate normal distribution and its applications in modeling.

CO2: Understand regression diagnostics and multivariate analysis of variance (MANOVA) to validate statistical models.

CO3: Apply the assumptions and estimation techniques used in multiple and multivariate regression models.

CO4: Analyze datasets using principal component analysis (PCA), factor analysis, and discriminant functions.

CO5: Evaluate and interpret clustering results using various clustering algorithms including k-means and hierarchical clustering.

CO-PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	2									
CO2	3	3	2	2							
CO3	3	3	3	3	2						
CO4	3	3	3	3	3						
CO5	3	3	3	3	3						

Course Content

Module 1: Multivariate Normal Distribution (6L)

Bivariate marginal, joint and conditional probability distributions. Multivariate normal distribution func- tions; conditional distributions and relation to regression; estimation of parameters.

Module 2: Multiple Linear Regression Model (7L)

Standard multiple regression models; diagnostics for collinearity, outliers, non-normality, autocorrelation; validation of assumptions.

Module 3: Multivariate Regression and MANOVA/MANCOVA (7L)

Assumptions of multivariate regression; parameter estimation; multivariate analysis of variance and covariance.

Module 4: Discriminant and Principal Component Analysis (8L)

Statistical background; linear discriminant function analysis and estimation. Principal components: algo- rithm, determining number of components, scree plot and H-plot.

Module 5: Factor and Cluster Analysis (8L)

Factor analysis models; extraction of common factors; determining number of factors; factor rotation and scoring. Cluster analysis: types of clustering, distance metrics, partitioning (k-means), hierarchical and overlapping clustering methods; interpretation of clusters.

Text Book:

1. **”Computational Statistics and Machine Learning”**, S. Pal and A. Chakroborty, Global Net Publication.

Reference Books:

1. **”An Introduction to Multivariate Statistical Analysis”**, T. W. Anderson, Wiley.
2. **”Multivariate Data Analysis”**, Hair, Black, Babin, Anderson, Pearson.
3. **”Classification and Clustering”**, Arabie, Hubert, and De Soete, Sage Publications.
4. **”Machine Learning with R”**, Brett Lantz, Packt Publishing.
5. **”Applied Multivariate Techniques”**, Subhash Sharma, Wiley.

Course Name: Computational Statistics

Course Code: M(CB)601

Contact (Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 3

Prerequisite:

Basic knowledge of probability, linear algebra, and statistical inference.

Course Objective:

The objective of the course is to make the students able to –

- Understand the theory and properties of multivariate normal distributions.
- Apply regression analysis in multivariate contexts and validate model assumptions.
- Analyze data using principal component and factor analysis techniques.
- Evaluate classification performance using discriminant analysis methods.
- Implement clustering algorithms and interpret cluster-based segmentation results.

Course Outcome(s):

After successful completion of the course, students will be able to:

CO1: Remember key concepts and properties of multivariate normal distribution and its applications in modeling.

CO2: Understand regression diagnostics and multivariate analysis of variance (MANOVA) to validate statistical models.

CO3: Apply the assumptions and estimation techniques used in multiple and multivariate regression models.

CO4: Analyze datasets using principal component analysis (PCA), factor analysis, and discriminant functions.

CO5: Evaluate and interpret clustering results using various clustering algorithms including k-means and hierarchical clustering.

CO-PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	2									
CO2	3	3	2	2							
CO3	3	3	3	3	2						
CO4	3	3	3	3	3						
CO5	3	3	3	3	3						

Course Content

Module 1: Multivariate Normal Distribution (6L)

Bivariate marginal, joint and conditional probability distributions. Multivariate normal distribution func- tions; conditional distributions and relation to regression; estimation of parameters.

Module 2: Multiple Linear Regression Model (7L)

Standard multiple regression models; diagnostics for collinearity, outliers, non-normality, autocorrelation; validation of assumptions.

Module 3: Multivariate Regression and MANOVA/MANCOVA (7L)

Assumptions of multivariate regression; parameter estimation; multivariate analysis of variance and covariance.

Module 4: Discriminant and Principal Component Analysis (8L)

Statistical background; linear discriminant function analysis and estimation. Principal components: algo- rithm, determining number of components, scree plot and H-plot.

Module 5: Factor and Cluster Analysis (8L)

Factor analysis models; extraction of common factors; determining number of factors; factor rotation and scoring. Cluster analysis: types of clustering, distance metrics, partitioning (k-means), hierarchical and overlapping clustering methods; interpretation of clusters.

Text Book:

1. **”Computational Statistics and Machine Learning”**, S. Pal and A. Chakroborty, Global Net Publication.

Reference Books:

1. **”An Introduction to Multivariate Statistical Analysis”**, T. W. Anderson, Wiley.
2. **”Multivariate Data Analysis”**, Hair, Black, Babin, Anderson, Pearson.
3. **”Classification and Clustering”**, Arabie, Hubert, and De Soete, Sage Publications.
4. **”Machine Learning with R”**, Brett Lantz, Packt Publishing.
5. **”Applied Multivariate Techniques”**, Subhash Sharma, Wiley.

Course Name: Compiler Design

Course Code: CB601

Contact (Periods/Weeks): 3 periods

Total Contact Hours: 36 hrs

Credits: 3

Prerequisites:

Knowledge of Data Structures and Algorithms, Knowledge of Automata Theory and Formal Languages, Basic programming skills in C/C++/Java

Course Objective(s):

The objective of the course is to make the students able to –

Understand the fundamental phases of a compiler and their interactions.

Apply lexical and syntax analysis techniques to analyze program code.

Design and implement parsers for context-free grammars.

Apply code optimization and intermediate code generation techniques.

Develop skills to design and implement a simple compiler for a programming language.

Course Outcomes (COs):

After successful completion of the course, students will be able to:

CO1: Explain the structure of a compiler and describe the role of each phase.

CO2: Apply lexical and syntax analysis techniques using tools like LEX and YACC.

CO3: Design parsers and translate context-free grammar into parse trees.

CO4: Generate intermediate code and apply optimization techniques.

CO5: Construct a simple compiler using modern compiler design tools and techniques.

CO–PO Mapping

CO–PO–PSO Mapping

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

Course Contents

Module 1: Introduction to Compilers [6L]

- Structure of a compiler, phases and their role
- Bootstrapping and compiler construction tools

Module 2: Lexical Analysis [6L]

- Role of lexical analyzer, specification of tokens
- Regular expressions, finite automata (DFA, NFA)
- Lexical analyzer generator (LEX)

Module 3: Syntax Analysis [8L]

- Context-free grammars, parse trees, ambiguity
- Top-down parsing (recursive descent, LL(1))
- Bottom-up parsing (shift-reduce, operator precedence, LR parsing, SLR, LALR)
- Parser generator (YACC)

Module 4: Semantic Analysis and Intermediate Code Generation [8L]

- Syntax-directed translation, attribute grammars
- Intermediate representations (three-address code, quadruples, triples)
- Symbol table management
- Type checking and error handling

Module 5: Code Optimization and Code Generation [8L]

- Code optimization: machine-independent and machine-dependent
- Peephole optimization, loop optimization
- Code generation: register allocation, instruction selection
- Runtime environment and storage organization

Learning Resources

Text Books:

1. Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman, *Compilers: Principles, Techniques, and Tools*, Pearson.
2. Santanu Chattopadhyay, *Compiler Design*, PHI Learning.

Reference Books:

1. Keith Cooper, Linda Torczon, *Engineering a Compiler*, Morgan Kaufmann.
2. Andrew W. Appel, *Modern Compiler Implementation in C*, Cambridge University Press.
3. John R. Levine, Tony Mason, Doug Brown, *Lex & Yacc*, O’Reilly Media.

Course Name: Mobile App Development

Course Code: CS602A

Contact (Periods/Week): 3

Total Contact Hours: 36

Credits: 3

Prerequisite:

Basic knowledge of object-oriented programming, database concepts, and computer networks.

Course Objective:

The objective of this course is to enable the students to:

- Understand the fundamentals of mobile computing and mobile application development.
- Learn the architecture, components, and life cycle of mobile applications.
- Develop skills in designing and developing mobile applications using modern frameworks and tools.
- Integrate mobile apps with databases, cloud, and other services for real-world use cases.

Course Outcome(s):

After the completion of the course students will be able to

CO1: Explain the concepts of mobile computing, architecture, and mobile application frameworks.

CO2: Design user interfaces for mobile applications considering usability and platform guidelines.

CO3: Develop mobile applications with data storage, retrieval, and connectivity features.

CO4: Integrate mobile applications with cloud services and APIs.

CO5: Deploy and test mobile applications in different platforms and environments.

CO-PO-PSO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO1	PSO1
CO1	3	2	2	2								2		
CO2	3	3	2	2	2							2	2	
CO3	3	2	3	3	2							3	2	2
CO4	2	2	3	3	3							2	3	2
CO5	3	3	3	3	3							3	3	3

Course Content

Module 1:

Introduction to Mobile Computing (6L)
Overview of mobile platforms, Android and iOS architecture, Mobile app development process.

Module 2:

Mobile Application Design (7L)
UI/UX principles, Activities, Fragments, Layouts, Views, Event handling, Material design guidelines.

Module 3:

Data Management in Mobile Apps (7L)
SQLite, Shared Preferences, Files, Content providers, Data synchronization.

Module 4:

Connectivity and Services (8L)
REST APIs, JSON, XML, Firebase, Cloud integration, Push notifications, Background services.

Module 5:

Deployment and Testing (8L)
App packaging, Deployment on Google Play and App Store, Debugging, Testing strategies, Case studies of real-world apps.

Text Books:

1. **”Professional Mobile Application Development”**, Jeff McWherter and Scott Gowell, Wrox.
2. **”Mobile Application Development”**, Pradeep Kothari, McGraw Hill Education.
3. **”Android Programming for Beginners”**, Joseph Anunchoria, Packt Publishing.

Reference Books:

(NIL)

Course Name: Computer Graphics with Agile Frameworks

Course Code: CB503A

Contact (Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 3

Prerequisite:

Programming Fundamentals: Proficiency in at least one programming language (preferably C/C++/Python), Basic Data Structures: Understanding of stacks, queues, matrices, and coordinate-based data, Mathematics: Familiarity with linear algebra, geometry, and trigonometry, Software Development Principles: SDLC, version control (Git), and OOP concepts, Discrete Mathematics (Recommended): Knowledge of logic, sets, and functions.

Course Objective:

The objective of this course is to enable the students to:

- Introduce the theoretical foundations and algorithms used in computer graphics.
- Explain geometric transformations, viewing, and projection in 2D and 3D.
- Describe modeling, lighting, shading, and rendering pipelines.
- Integrate Agile concepts like Scrum, backlogs, and iterative delivery into graphics projects.
- Expose students to convergence of graphics and agile execution strategies.

Course Outcome(s):

After completion of the course students will be able to

CO1: Explain key concepts and illustrate algorithms in computer graphics.

CO2: Apply geometric transformations and construct viewing pipelines in 2D and 3D space.

CO3: Analyze rendering pipelines and differentiate among shading techniques.

CO4: Design graphics system architectures and evaluate their components.

CO5: Describe Agile principles and demonstrate planning techniques for iterative graphics projects.

CO-PO-PSO Mapping

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

Course Content

Module 1: Foundations of Computer Graphics (7L)

Introduction and applications of computer graphics. Raster vs random scan systems. Graphics pipeline overview. Line, circle, and ellipse drawing algorithms: DDA, Bresenham. Scan conversion and pixel plotting techniques.

Module 2: 2D and 3D Geometric Transformations (7L)

2D transformations: translation, rotation, scaling, shearing, reflection. Homogeneous coordinates and matrix representations. Composition of transformations. 3D transformations: scaling, rotation about principal axes, translation. Perspective and orthographic projections.

Module 3: Viewing, Clipping and Hidden Surface Removal (7L)

Viewing pipeline and coordinate systems. Windowing, viewport, and mapping. Clipping algorithms: Cohen-Sutherland, Liang-Barsky, Sutherland-Hodgman. Hidden surface removal techniques: Z-buffer, Painter's algorithm.

Module 4: Curves, Shading, and Rendering (7L)

Curve representation: Bezier, B-splines, Hermite curves. Basic illumination models: ambient, diffuse, spec- ular. Shading techniques: flat, Gouraud, Phong. Rendering pipeline: modeling, lighting, viewing, rasteriza- tion. Texture mapping, anti-aliasing.

Module 5: Agile Frameworks in Graphics Project Planning (8L)

Agile principles and Scrum methodology. Agile roles: Product Owner, Scrum Master, Dev Team. User stories, backlog, sprint planning, burn-down charts. Incremental graphics software development. Integrating Agile with graphics project milestones.

Text Books:

1. **"Computer Graphics with OpenGL"**, Donald Hearn and M. Pauline Baker, Pearson Education, 4th Edition.
2. **"Interactive Computer Graphics: A Top-Down Approach with WebGL"**, Edward Angel and Dave Shreiner, Addison-Wesley, 7th Edition.
3. **"Fundamentals of Computer Graphics"**, Peter Shirley, CRC Press, 5th Edition.

Reference Books:

1. **"Computer Graphics: Principles and Practice"**, J.D. Foley, A. van Dam, S.K. Feiner, and J.F. Hughes, Addison-Wesley, 3rd Edition.
2. **"Fundamentals of Computer Graphics"**, Peter Shirley, CRC Press, 5th Edition.
3. **"Computer Graphics: A Programming Approach"**, Steven Harrington, McGraw-Hill, 2nd Edition.
4. **"The Scrum Guide"**, Ken Schwaber and Jeff Sutherland, Scrum.org, Latest Edition.
5. **"User Stories Applied: For Agile Software Development"**, Mike Cohn, Addison-Wesley, 1st Edition.

3rd Year 6th Semester									
Sl. No.	Broad Category	Category	Paper Code	Subject	L	T	P	Total	Credit Points
A. Theory									
1	SCI	Multi	M(CB)601	Computational Statistics	3	0	0	3	3
2	ENGG	Major	CB601	Compiler Design	3	0	0	3	3
3	ENGG	Major	CB602A	Mobile App Development	3	0	0	3	3
			CB602B	Introduction to Internet of Things					
			CB602C	Enterprise Resource Planning (ERP) Systems					
			CB602D	Digital Image Processing					
4	ENGG	Minor	HU(CB)601A	Entrepreneurship and Startup Ecosystem	3	0	0	3	3
			HU(CB)601B	Principles of Marketing in the Digital Era					
			HU(CB)601C	Business Communication and Soft Skills					
			HU(CB)601D	Intellectual Property Rights and Patents					
5	SCI	Multi	M(CB)602	Operation Research	2	0	0	2	2
6	HUM	Minor	BS601	Human Resource Management	3	0	0	3	3
B. Practical									
1	SCI	Multi	M(CB)691	Computational Statistics Lab with R Programming	0	0	3	3	1.5
2	ENGG	Major	CB692A	Mobile App Development Lab	0	0	3	3	1.5
			CB692B	Introduction to Internet of Things (IoT) Lab					
			CB692C	Enterprise Resource Planning (ERP) Systems Lab					
			CB692D	Digitial Image Processing Lab					
3	PRJ	PRJ	CB681	Project-II	0	0	8	8	4
C. Mandatory Activities / Courses									
1	MC	Mandatory Course	MC681	Course from Skill India-III*	0	0	0	0	2 Units
Total of Theory, Practical								31	24

Course Name: Introduction To Internet Of Things

Course Code: CB602B

Contact (Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 3

Prerequisite:

Basic knowledge of electronics, computer networks, and embedded systems.

Course Objective:

The objective of this course is to enable the students to:

- Understand the fundamental concepts of IoT and differentiate between Consumer IoT and Industrial IoT.
- Explore IoT reference architectures, edge computing, gateways, and data pipelines.
- Learn sensor integration, industrial systems, and data acquisition in real-world scenarios.
- Investigate IoT communication protocols, networking layers, and web-based interactions.
- Analyze IoT data storage, processing, visualization, and security mechanisms.

Course Outcome(s):

After the completion of the course students will be able to

CO1: Explain the basic concepts, use cases, and industrial relevance of IoT.

CO2: Describe IoT architecture, edge devices, and data processing pipelines.

CO3: Integrate sensors and understand industrial control and acquisition systems.

CO4: Apply communication protocols and networking for IoT systems.

CO5: Analyze IoT data characteristics and perform basic analytics and visualization.

CO-PO-PSO Mapping

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

Course Content

Module I:

Introduction to IoT and Use Cases (6L)

1.01 Fundamental concepts of IoT

1.02 Consumer IoT vs Industrial Internet

1.03 Fundamental building blocks of IoT

1.04 Industry use cases: smart cities, healthcare, agriculture, logistics

Module II:

IoT Architecture and Data Pipelines (6L)

2.01 IoT reference architecture

2.02 Industrial Internet Reference Architecture

2.03 Edge computing and IoT gateways

2.04 Data ingestion and stream processing

Module III:

Sensors and Industrial Systems (6L)

3.01 Basics of sensors and transducers

3.02 Sensor integration with processing boards

3.03 Industrial data acquisition systems: SCADA, PLC

3.04 Industrial control system functions

Module IV:

Networking and Communication for IoT (6L)

4.01 OSI model and IoT mapping

4.02 Proximity technologies: ZigBee, Bluetooth, Serial Communication

4.03 Industrial protocols: Modbus, CANbus

4.04 Cloud protocols: REST, TCP/IP, UDP, MQTT, WebSockets

4.05 Message encoding: JSON, Protocol Buffers

Module V:

IoT Data Processing and Storage (6L)

5.01 Time series data and databases

5.02 Basic time series analytics

5.03 Data summarization, sketching, handling noise/missing data

5.04 Anomaly and outlier detection

Optional Activities:

- Smart City & Smart Home Applications
- LPWAN Technologies
- Secure Communication in IoT
- IoT Platform Comparisons
- IoT and Edge AI
- Cloud & Fog Computing in IoT
- IoT Privacy and Regulation

Text Book:

1. **”The Internet of Things”** – Samuel Greengard, MIT Press Essential Knowledge Series.

Reference Books / Online Resources:

1. **”Industrial Internet Reference Architecture”**
2. **”WEF Report on Industrial IoT”**
3. **”50 Sensor Applications for a Smarter World”** – Libelium
4. **”Visualizing Data”** – Ben Fry, O’Reilly Media
5. **”Raspberry Pi Computer Architecture Essentials”** – Andrew K Dennis
6. **”Getting Started with Arduino”** – M. Banzi, O’Reilly Media
7. **”GSMA IoT Security Guidelines”**

Course Name: Enterprise Resource Planning Systems

Course Code: CB602C

Contact (Periods/Week): 3

Total Contact Hours: 36

Credits: 3

Prerequisite:

Basic IT knowledge and understanding of business processes.

Course Objective:

The objective of the course is to make the students able to –

- Understand ERP concepts, architecture, and evolution.
- Learn about different ERP modules and business integration.
- Analyze ERP implementation and lifecycle.
- Explore case studies and ERP success/failure factors.
- Understand ERP market trends and emerging technologies.

Course Outcome(s):

After successful completion of the course, students will be able to

CO1: Describe ERP fundamentals, architecture, and business process integration.

CO2: Compare ERP modules and evaluate their organizational impact.

CO3: Apply ERP selection and implementation strategies.

CO4: Analyze real-world ERP cases and identify risk factors.

CO5: Evaluate current ERP software and market trends.

CO-PO-PSO Mapping

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

Course Content

Module 1:

Introduction to ERP (6L)
Need, evolution, components, architecture, business integration.

Module 2:

ERP Modules (8L)
Finance, HR, Manufacturing, Sales & Distribution, SCM, CRM.

Module 3:

ERP Implementation (6L)
Lifecycle, process modeling, tools, success/failure factors.

Module 4:

ERP Market (8L)
SAP, Oracle, Microsoft Dynamics, Tally, open-source ERP.

Module 5:

Emerging Trends (8L)
Cloud ERP, Mobile ERP, AI in ERP, integration with IoT.

Text Books:

1. **”Enterprise Resource Planning”** – Alexis Leon, McGraw-Hill, Current Edition.
2. **”Concepts in ERP”** – Monk & Wagner, Cengage Learning.

Reference Books:

1. **”ERP Demystified”** – Alexis Leon, McGraw-Hill.
2. **”ERP: Making it Happen”** – Thomas Wallace, Wiley.
3. **”Modern ERP”** – Marianne Bradford, Lulu Press.

Course Name: Digital Image Processing

Course Code: CB602D

Contact (Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 3

Prerequisite:

Basic Image Processing idea.

Course Objective:

The objective of this course is to enable the students to:

- Introduce the fundamental principles and mathematical models of digital image processing.
- Develop knowledge of image enhancement, restoration, and compression methods.
- Enable students to implement image segmentation, representation, and object recognition techniques.
- Provide hands-on skills for applying image processing concepts to real-world applications.

Course Outcome(s):

After completion of the course students will be able to:

CO1: Understand the fundamental concepts and mathematical models of digital image processing.

CO2: Apply spatial and frequency domain enhancement techniques for image improvement.

CO3: Implement segmentation and feature extraction methods for image analysis.

CO4: Analyze and evaluate image restoration and compression techniques.

CO5: Design and develop real-world applications using digital image processing algorithms.

CO-PO-PSO Mapping

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

Course Content

Module I: Introduction

Fundamentals of digital image processing, image representation, sampling and quantization. Basic relationships between pixels, image formation models, applications of DIP.

Module II: Image Enhancement

Spatial domain techniques: contrast stretching, histogram equalization, smoothing and sharpening filters. Frequency domain enhancement: Fourier transform, low-pass and high-pass filtering, homomorphic filtering.

Module III: Image Restoration and Compression

Image degradation models, noise models, inverse filtering, Wiener filtering. Image compression: coding redundancy, transform coding, JPEG, MPEG standards.

Module IV: Image Segmentation and Representation

Edge detection, thresholding, region growing, region splitting and merging. Watershed algorithm, morphological segmentation. Boundary and regional descriptors.

Module V: Object Recognition and Applications

Pattern and object recognition basics. Applications: medical imaging, biometrics, remote sensing, computer vision. Emerging trends in image analysis.

Text Books:

1. **”Digital Image Processing”**, Rafael C. Gonzalez and Richard E. Woods, 4th Edition, Pearson.
2. **”Fundamentals of Digital Image Processing”**, Anil K. Jain, PHI Learning.

Reference Books:

1. **”Digital Image Processing and Analysis”**, B. Chanda and D. Dutta Majumder, PHI.
2. **”Digital Image Processing”**, Kenneth R. Castleman, Pearson.

Course Name: Entrepreneurship and Startup Ecosystem

Course Code: HU(CB)601A

Contact (L:T:P): 0:0:3

Total Contact Hours: 34

Credits: 2

Prerequisites:

- None

Course Objective(s):

The objective of the course is to make the students able to –

O1: Understand the fundamentals of entrepreneurship and entrepreneurial mindset.

O2: Analyze decision-making processes in entrepreneurial ecosystems.

O3: Develop and evaluate business models and lean startup strategies.

O4: Comprehend funding structures and entrepreneurial finance mechanisms.

O5: Explore innovation and social entrepreneurship in the Indian and global context.

Course Outcomes (COs):

After successful completion of the course, students will be able to:

CO1: Comprehend the role of bounded rationality, framing, causation in entrepreneurial decision-making.

CO2: Explain the fundamentals of developing and presenting business pitching to potential investors.

CO3: Apply business model development, financial planning and resource mobilization strategies to design effective startup plans.

CO4: Evaluate the various sources of raising finance for startup ventures.

CO5: Demonstrate an ability to design a business model canvas.

CO–PO Mapping

CO–PO–PSO Mapping

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

Course Contents

Module 1: Introduction to Entrepreneurship [4L]

- Entrepreneurs; entrepreneurial personality and intentions – characteristics, traits and behavioral aspects
- Entrepreneurial challenges and opportunities
- Value creation; prototype and exemplar models; reverse engineering

Module 2: Entrepreneurial Process and Decision Making [6L]

- Ecosystem, ideation, opportunity exploitation
- Negotiation, effectuation vs causation
- Limitations of entrepreneurship

Module 3: Crafting Business Models and Lean Startups [8L]

- Conventional vs innovation logic
- Value proposition design; business model canvas
- Lean startups; business pitching

Module 4: Organizing Business and Entrepreneurial Finance [10L]

- Business structures, evolution of organizations
- Venture finance options, policy initiatives
- Role of institutions

Module 5: Entrepreneurs as Problem Solvers [6L]

- Innovation and entrepreneurship – global and Indian perspectives
- Technology and e-commerce
- Social entrepreneurship; Indian ecosystem

Learning Resources

Text Books:

1. Rajeev Roy, *Entrepreneurship*, Oxford University Press.
2. Peter F. Drucker, *Innovation and Entrepreneurship*, HarperBusiness.

Reference Books:

1. J.A. Timmons & S. Spinelli, *New Venture Creation*, McGraw-Hill.
2. Eric Ries, *The Lean Startup*, Crown Publishing.
3. A. Osterwalder & Y. Pigneur, *Business Model Generation*, Wiley.

Course Name: Principles of Marketing in the Digital Era

Course Code: HU(CB)601B

Contact (Periods/Week): 0:0:3

Total Contact Hours: 34

Credits: 3

Prerequisite:

NIL

Course Objective:

The objective of the course is to make the students able to –

- Acquaint the students with the knowledge of growing integration between the traditional and digital marketing concepts and practices in the digital era.
- Familiarize the students with the tools and techniques used by the digital marketers for driving the marketing decisions to attain marketing objectives.

Course Outcome(s):

After successful completion of the course, students will be able to

CO1: Understand the concept of digital marketing and its integration with traditional marketing.

CO2: Understand customer value journey in digital context and behaviour of online consumers.

CO3: Understand email, content and social media marketing and apply the learning to create digital media campaigns.

CO4: Examine various tactics for enhancing a website’s position and ranking.

CO5: Leverage digital strategies to gain competitive advantage for business.

CO-PO-PSO Mapping

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

Course Content

Module 1:

Introduction to Marketing & Principles of Marketing (6L)
4Ps vs 4Cs, The Marketing Process and Challenges in the New Economy, Defining and measuring customer profitability, Defining Customer Value & Satisfaction.

Module 2:

Competitive Markets (6L)
Competitive markets and Market attractiveness, Competitive Intelligence Systems, Competitive Strategies: Leaders, Followers, Niche, Challengers, Developing, Differentiating and Positioning Products through Prod- uct Life Cycle.

Module 3:

Market Segmentation and Branding (6L)
Market Segmentation and Targeting Strategies, Managing Product Lines and Brands, Brand Equity and Labelling: to brand or not to brand?

Module 4:

Marketing Channels and Communication (6L)
Selecting and managing marketing channels, Marketing Communications Mix, Promotion and Advertising Strategies, Pricing strategies.

Module 5:

Digital Marketing in the New Era (10L)
The New “e-nvironment” & Key Internet Marketing Concepts, Internet User Characteristics & Behavior, Marketing Channels, Market Structure and the Internet, Word-of-mouth communications (positive vs negative WOM), Opinion leadership in Digital modes: the power of online communities, A New Marketing Model for the Digital Age, Creating a Knowledge Organization: Customer Insight & Segmentation, E-Marketing Strategies: Communication, Product, Pricing & Website Promotion Strategies.

Text Books:

1. **”Principles of Marketing- A South Asian Perspective”**, Kotler, Armstrong, Agnihotri and Haque, 13th edition, Pearson Education, 2010.
2. **”Marketing Management”**, Rajan Saxena, 3rd edition, McGraw Hill Education, 2008.

Reference Books:

1. **”The Art of Digital Marketing”**, Ian Dodson, Wiley, 2016.
2. **”Marketing 4.0: Moving from Traditional to Digital”**, Hermawan Kartajaya, Philip Kotler, Iwan Setiawan, Wiley, 2016.
3. **”Digital Marketing”**, Seema Gupta, McGraw Hill.

Course Name: Business Communication and Soft Skills

Course Code: HU(CB)601C

Contact (Periods/Week): 3:0:0

Total Contact Hours: 48

Credits: 3

Prerequisite:

Basic knowledge of Professional Communication and high school grammar.

Course Objective:

The objective of the course is to make the students able to –

- Understand the importance of corporate communication.
- Practice corporate etiquettes in real life scenarios.
- Recognize soft skills as an indispensable part of the business world.
- Develop reading proficiency skills and enhance verbal ability.
- Apply group behavioral skills, group discussion strategies, and personal interview techniques.

Course Outcome(s):

After successful completion of the course, students will be able to:

CO1: Apply essential interpersonal skills including teamwork, time management, leadership, and conflict resolution in academic and professional contexts.

CO2: Demonstrate effective use of verbal and non-verbal communication in diverse professional scenarios.

CO3: Solve verbal ability questions including analogies, synonyms, sentence correction, and para jumbles with improved reading speed and vocabulary.

CO4: Prepare and deliver structured technical presentations and professional technical reports.

CO5: Participate confidently and effectively in group discussions and personal interviews, displaying appropriate strategies.

CO-PO-PSO Mapping

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

Course Content

Module 1:

Introduction to Soft Skills

- The skills of interpersonal communication
- Team behavior
- Time management skills
- Decision making and negotiation skills
- Conflict resolution
- Leadership skills

Module 2:

Application of Non-Verbal Communication in Professional Sphere

- Types of non-verbal communication
- Significance of non-verbal communication in the corporate world
- Practice through role plays

Module 3:

Verbal Ability Test Patterns

- Verbal ability tests: synonyms, antonyms, error spotting/sentence improvement
- Analogies and para jumbles
- Reading comprehension
- Enhancing reading speed and vocabulary

Module 4:

Technical Presentation and Technical Report Writing

- Rules for effective technical presentation
- Assignments on presentations
- Writing a technical report
- Report formats and practice sessions

Module 5:

Group Discussion and Personal Interview

- Basics of group discussion and intensive practice
- Case study-based discussions
- Types of interviews
- Strategies for answering interview questions
- Mock interviews

Text Books:

1. **”How to Develop Self Confidence and Improve Public Speaking”** – Dale Carnegie.
2. **”Professional English in Use”** – Mark Ibbotson, Cambridge, 2009.

Reference Books:

1. **”Technical Communication”** – Meenakshi Raman and Sangeetha Sharma, 3rd Edition, Oxford University Press, 2015.
2. **”Cambridge English for Engineering”** – Mark Ibbotson, Cambridge University Press, 2008.
3. **”Writing Reports”** – John Seeley, Oxford University Press, 2002.
4. **”E-writing: 21st Century Tools for Effective Communication”** – Diana Booher, Macmillan, 2007.
5. **”Practical English Usage”** – Michael Swan, Oxford University Press, 1980.

Course Name: Intellectual Property Rights and Patents

Course Code: HU(CB)601D

Contact (Periods/Week): 0:0:3

Total Contact Hours: 34

Credits: 3

Prerequisite:

None

Course Objective:

The objective of the course is to make the students able to –

- Understand the concept and importance of Intellectual Property Rights (IPR).
- Gain awareness of various IPR categories such as patents, copyrights, trademarks, and geographical indications.
- Learn the procedures and legal frameworks for filing and protecting IPR in India and abroad.
- Explore national and international institutions and laws governing IPR.
- Examine recent trends, policies, and emerging challenges in the field of IPR.

Course Outcome(s):

After successful completion of the course, students will be able to

CO1: Explain fundamental aspects of Intellectual Property Rights to students.

CO2: Disseminate knowledge on patents and patent regimes in India and abroad.

CO3: Disseminate knowledge on copyrights and related rights and registration.

CO4: Disseminate knowledge on trademarks, design, GI, and layout design protection.

CO5: Be aware of current trends and government steps in fostering IPR.

CO-PO-PSO Mapping

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

Course Content

Module 1:

Overview of IPR (4L)

Introduction and the need for IPR. Kinds of IPR: Patents, Copyright, Trade Mark, Design, GI, Plant Variety, Layout Design. Genetic Resources and Traditional Knowledge. Trade Secret. IPR in India and abroad. International organizations and treaties.

Module 2:

Patents (4L)

Definition, kinds of inventions, elements of patentability. Legal requirements. Patent application process: Searching, drafting, filing. Rights and duties. Restoration, infringement, remedies.

Module 3:

Trademarks (6L)

Concept, types of marks. Registration process. Rights, assignment, licensing, infringement, remedies.

Module 4:

Copyrights (6L)

Law of copyrights. Ownership, registration, infringement, penalties. Cyber law, privacy, freedom of expression, net neutrality, national security.

Module 5:

GI and Industrial Design (8L)

Types, why GI needs protection, GI laws. Industrial design protection. Integrated circuits.

Module 6:

National IPR Policy 2016 (6L)

Govt. initiatives, career opportunities, current trends in IPR, case studies.

Text Books:

1. **”Fundamentals of IPR for Engineers”** – K. Bansal & P. Bansal, Tata McGraw Hill
2. **”Intellectual Property Rights: Protection and Management”** – Neeraj Pandey & Khushdeep Dhillon, PHI Learning

Reference Books:

1. **”Law relating to Intellectual Property Rights”** – V.K. Ahuja, Lexis Nexis, 2017.
2. **”Intellectual Property Rights”** – K.V. Nithyananda, Cengage Learning, 2019.
3. **”Principles of Marketing”** – Philip Kotler and Gary Armstrong, Pearson Publications.

Course Name: Operations Research

Course Code: M(CB)602

Contact (Periods/Week): 2:0:0

Total Contact Hours: 24

Credits: 2

Prerequisite:

Basic understanding of algebra, probability, and matrix operations.

Course Objective(s):

The objective of the course is to make the students able to:

- O1: Understand the fundamental principles and phases of operations research (OR).
- O2: Formulate and solve linear programming, transportation, and assignment problems.
- O3: Apply network-based project scheduling techniques.
- O4: Analyze inventory and queuing systems using appropriate models.
- O5: Simulate discrete event systems for scheduling and resource optimization.

Course Outcome(s):

After successful completion of the course, students will be able to:

CO1: Remember basic terminology, model types, and foundational concepts in Operations Research.

CO2: Understand the formulation process of linear, transportation, and assignment problems.

CO3: Apply graphical, simplex, and MODI methods to solve OR models and interpret results.

CO4: Analyze project networks (PERT/CPM), queuing models for decision-making.

CO5: Evaluate inventory systems and system performance using simulation techniques and compare different policies.

CO-PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	2									
CO2	3	3	2								
CO3	3	3	3	2							
CO4	3	3	3	3	2						
CO5	3	3	3	3	3						

Course Content

Module 1: Introduction to OR (3L)

Origin and definition of OR; optimization performance measures; deterministic vs. stochastic models; OR problem-solving phases.

Module 2: Linear Programming (5L)

Formulation of LPP; matrix representation; concepts from linear algebra – vectors, rank, independence, hyperplanes, convexity. Geometric method (2 variables); simplex method; special cases (degeneracy, unboundedness, redundancy); duality and dual simplex.

Module 3: Transportation & Assignment Problems (5L)

Formulation and solution of transportation problems using NWCR, VAM, and MODI method. Assignment problems and Hungarian method. Handling of unbalanced problems and degeneracy.

Module 4: Project Scheduling (PERT–CPM) (4L)

Project networks, critical paths, project time and variance estimation using PERT, project crashing and time-cost trade-off.

Module 5: Inventory & Queuing Theory (4L)

EOQ, POQ models, quantity discount models, stockouts and safety stocks, ABC analysis. Queuing models – characteristics, Kendall's notation, M/M/1 and M/M/m models, Little's law.

Module 6: Simulation Methodology (3L)

Definition and applications; steps of simulation; discrete event simulation, event list, use in scheduling, queuing, and inventory systems.

Text Book:

1. **"Operations Research: An Introduction"**, H. A. Taha.

Reference Books:

1. **"Linear Programming"**, K. G. Murthy.
2. **"Linear Programming"**, G. Hadley.
3. **"Principles of OR with Application to Managerial Decisions"**, H. M. Wagner.
4. **"Introduction to Operations Research"**, F. S. Hiller and G. J. Lieberman.
5. **"Elements of Queuing Theory"**, Thomas L. Saaty.
6. **"Operations Research and Management Science Handbook"**, A. Ravi Ravindran.
7. **"Management Guide to PERT/CPM"**, Wiest & Levy.
8. **"Modern Inventory Management"**, J. W. Prichard and R. H. Eagle.

Course Name: Mobile App Development Lab

Course Code: CB692A

Contact (Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 1.5

Prerequisite:

Basic knowledge of object-oriented programming, databases, and mobile app frameworks.

Course Objective:

The objective of this course is to enable the students to:

- Provide practical exposure to mobile application design and development.
- Develop hands-on skills in creating Android/iOS applications using modern tools.
- Enable students to integrate mobile applications with local databases and cloud services.
- Familiarize students with testing, debugging, and deployment of mobile apps.
- Encourage innovation through mini-projects and real-world app development use cases.

Course Outcome(s):

After completion of the course students will be able to:

CO1: Install and configure mobile app development environments and build basic applications.

CO2: Design and implement user interfaces for mobile applications.

CO3: Develop apps with data storage, retrieval, and synchronization features.

CO4: Integrate mobile apps with cloud platforms, APIs, and services.

CO5: Test, debug, and deploy mobile applications on different platforms.

CO-PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	3	2	2							2	
CO2	3	3	2	2						2	
CO3	3	2	3	3	2					2	2
CO4	3	2	3	3	3					3	2
CO5	3	3	3	3	3	2				3	3

List of Suggested Experiments

- Setting up Android Studio / Xcode and creating a “Hello World” mobile application.
- Designing user interfaces using layouts, views, and event handling.
- Implementing activities, fragments, and navigation in mobile apps.
- Data persistence with SQLite and Shared Preferences.
- Using content providers and files for local data management.
- Consuming REST APIs with JSON and XML parsing.
- Integrating Firebase for authentication and real-time database operations.
- Implementing cloud storage and push notifications.
- Creating background services and broadcast receivers.
- Packaging, testing, debugging, and deployment of mobile apps.
- Mini project: Full mobile application development (individual or group).

Text Books:

1. **”Professional Mobile Application Development”**, Jeff McWherter and Scott Gowell, Wrox.
2. **”Mobile Application Development”**, Pradeep Kothari, McGraw Hill Education.
3. **”Android Programming for Beginners”**, Joseph Anunchoria, Packt Publishing.

Reference Books:

(NIL)

Course Name: Human Resource Management

Course Code: BS601

Contact (Periods/Week): 3

Total Contact Hours: 36

Credits: 3

Prerequisite:

Students must have a basic understanding of Project Management principles.

Course Objective(s):

- Understand fundamental frameworks and principles of HRM and its significance in business.
- Learn how to manage human resources effectively and in compliance with legal frameworks.
- Evaluate HR planning, development, and compensation systems.
- Analyze real-world HR issues including motivation, grievances, and industrial relations.
- Apply HRM concepts to improve organizational productivity and employee satisfaction.

Course Outcome(s):

After successful completion of the course, students will be able to:

CO1: Explain the importance of human resources and their effective management in organizations.

CO2: Apply tools in managing employees effectively and ensure compliance with regulations.

CO3: Analyze key issues in motivation, compensation, training, diversity, and ethics.

CO4: Evaluate different tools used in planning and maintaining HR needs.

CO5: Demonstrate HRM strategies that promote teamwork, leadership and continuous professional development.

CO-PO-PSO Mapping

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

Course Content

Module 1: Introduction (4L)

Definition, scope, importance, functions of HRM and its role in managerial functions. Role and position of personnel function in the organization.

Module 2: Procurement and Placement (3L)

Need and process of HR planning, recruitment, psychological tests, interviews, placement, and induction. Important Acts: Employment Exchange Act 1959, The Contract Labour Act 1970.

Module 3: Training & Development (3L)

Training vs. development, principles of training, employee development, merit vs. seniority, career development and performance appraisal.

Module 4: Job Analysis & Design (2L)

Job analysis, job description, job specification.

Module 5: Job Satisfaction (8L)

Motivation theories, quality of work life, workers' participation. Wage policies: compensation function, issues in administration, bonuses, incentives, job evaluation. Relevant Acts: Payment of Wages Act 1936, Minimum Wages Act 1948.

Module 6: Industrial Relations (8L)

Human and industrial relations, grievance handling, absenteeism, turnover, employee-employer relationship, discipline, industrial disputes, collective bargaining.

Module 7: Maintenance (8L)

Fringe and retirement benefits, welfare amenities, employee safety and security. Statutory provisions and Acts: Factories Act 1948, ESI Act 1948, Gratuity Act 1972, Social Security, Family Pension Scheme. Chal- lenges and future directions in HRM.

Text Book:

1. **"Human Resource Management"**, T. N. Chhabra, Dhanpat Rai & Co.

Reference Books:

1. **"Principles of Personnel Management"**, Edwin B. Flippo, McGraw-Hill.
2. **"Labour Problems and Social Welfare"**, R. C. Saxena, K. Math & Co.
3. **"Personnel Management"**, M. S. Saiyadain, Tata McGraw-Hill.
4. **"Personnel Management"**, C. B. Mamoria, Himalaya Publishing House.

Course Name: Introduction to Internet of Things (IoT) Lab

Course Code: CB692B

Contact (Periods/Week): 0:0:3

Total Contact Hours: 36

Credits: 3

Prerequisite:

Basic knowledge of microcontrollers, sensors, and programming (Python/C/C++ preferred).

Course Objective:

The objective of the course is to make the students able to –

- Develop a practical understanding of IoT architecture using sensors, actuators, and microcontrollers.
- Integrate various sensors and actuators with platforms such as ESP32, Arduino, and Raspberry Pi.
- Transmit and visualize sensor data using standard communication protocols like HTTP and MQTT.
- Implement real-time IoT applications using cloud platforms and mobile-based control systems.
- Explore UAV systems and their integration with IoT-based applications.

Course Outcome(s):

After successful completion of the course, students will be able to:

CO1: Interface and control actuators like LED, servo motors, and relays using ESP32/Arduino.

CO2: Integrate multiple sensors (ultrasonic, LDR, PIR, DHT) to build real-time embedded systems.

CO3: Design and deploy IoT systems using Raspberry Pi and cloud services like ThingSpeak and Blynk.

CO4: Develop web-based and app-based control systems using HTTP and MQTT protocols.

CO5: Understand and simulate UAV-based IoT applications with flight controller and GPS modules.

CO-PO-PSO Mapping

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

Course Content

List of Experiments

Using ESP32 / Arduino:

- Actuator Control via Serial Monitor: LED patterns, button switch control, servo motor with joystick
- Distance Measurement: Ultrasonic sensor interfacing with LCD for object detection

- Sensor-Based Automation:
  - Relay control using LDR
  - Burglar alarm using PIR & buzzer
  - Temperature and humidity display using DHT & LCD

Using Raspberry Pi / Arduino:

- Advanced Sensor Interfacing:
  - Relay via IR sensor
  - Stepper motor control
  - Burglar alarm with PIR, buzzer, and keypad
  - Smart lighting with PIR and LDR combination

IoT Frameworks and Cloud Integration:

- ThingSpeak Logging: Uploading humidity and light level data periodically
- Mobile App Control: Controlling LEDs, relays, buzzers using the Blynk platform

HTTP-Based Control:

- Basic Web Hosting on ESP32: Web interface for controlling actuators
- Web Display of Sensor Readings: Hosting real-time temperature/humidity on ESP32 webpage

MQTT-Based Applications:

- Remote Device Control: LED/Motor and AC appliance control via web/mobile app
- Sensor Data Dashboard: Publish sensor data like humidity/temperature to a web-based interface

UAV / Drone-Based IoT Systems:

- UAV Introduction and Planning: Drone components and flight mission planning using Mission Planner
- GPS Data Acquisition: Python program to read GPS coordinates from UAV flight controller

Course Name: Enterprise Resource Planning (ERP) Systems Lab

Course Code: CB692C

Contact (Periods/Week): 3

Total Contact Hours: 30

Credits: 1.5

Prerequisite:

None

Course Objective:

The objective of the course is to make the students able to –

- Gain practical hands-on exposure to ERP environments.
- Simulate business processes using ERP software.
- Create functional ERP modules for basic tasks.
- Generate and interpret ERP reports.
- Integrate data and processes using ERP platforms.

Course Outcome(s):

After successful completion of the course, students will be able to

CO1: Navigate ERP interface and basic operations.

CO2: Demonstrate data entry and process management in modules.

CO3: Customize standard ERP workflows.

CO4: Interpret ERP-generated business reports.

CO5: Simulate end-to-end transaction cycles in ERP.

CO-PO-PSO Mapping

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

Course Content

Module 1:

ERP Introduction and Navigation (6L)
Overview of ERP systems and navigating ERP environment.

Module 2:

Sales and Distribution Processes; Purchase and Inventory Management (8L)
Hands-on practice with sales orders, purchase orders, and inventory tracking.

Module 3:

Financial Accounting and Reporting (8L)
ERP financial accounting workflows, generating and analyzing reports.

Module 4:

Payroll and HR Modules (8L)
Simulating HR processes such as payroll management, employee records, and reporting.

Text Books:

1. **”ERP Lab Manual”**, Custom Institute Edition.
2. **”SAP ERP: A Beginner’s Guide”**, Biao Fu, McGraw Hill.

Reference Books:

1. **”Learning SAP ERP”**, Surya Padhi, Packt Publishing.
2. **”ERP Training Manual”**, Online/Custom Material.
3. **”ERP Tutorials and Labs”**, Cengage.

Course Name: Digital Image Processing Lab

Course Code: CB692D

Contact (Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 1.5

Prerequisite:

Basic programming knowledge.

Course Objective:

The objective of this course is to enable the students to:

- Provide practical knowledge of digital image processing techniques through laboratory experiments.
- Enable students to implement and analyze image enhancement, filtering, and restoration methods.
- Impart hands-on skills for image segmentation, feature extraction, and compression.
- Encourage the application of digital image processing tools to solve real-world problems.
- Promote innovation through mini-projects and real-time applications in medical imaging, biometrics, and computer vision.

Course Outcome(s):

After completion of the course students will be able to:

CO1: Perform basic operations on digital images using software tools.

CO2: Apply image enhancement and filtering techniques to improve image quality.

CO3: Implement noise models, restoration, and compression algorithms.

CO4: Analyze segmentation and morphological methods for object recognition.

CO5: Design and develop mini-projects using digital image processing concepts.

CO-PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11
CO1	2	2									
CO2	2	2	1							1	
CO3	2	2	1	1						2	1
CO4	2	2	2	2	2					2	1
CO5	2	2	1	1	1	2				2	1

List of Suggested Experiments

- Reading, writing, and displaying images.
- Histogram plotting and histogram equalization.
- Contrast stretching and intensity transformations.
- Image smoothing with averaging and Gaussian filters.
- Sharpening with Laplacian and high-pass filters.
- Frequency domain filtering using Fourier transform.
- Adding and removing different types of noise.
- Restoration using inverse and Wiener filtering.
- Edge detection: Sobel, Prewitt, and Canny.
- Segmentation: thresholding and region growing.
- Morphological operations: erosion, dilation, opening, closing.
- Image compression using JPEG basics.
- Mini project on real-world applications.

Text Books:

1. **"Digital Image Processing"**, Rafael C. Gonzalez and Richard E. Woods, 4th Edition, Pearson.
2. **"Fundamentals of Digital Image Processing"**, Anil K. Jain, PHI Learning.

Reference Books:

1. **"Digital Image Processing and Analysis"**, B. Chanda and D. Dutta Majumder, PHI.
2. **"Digital Image Processing"**, Kenneth R. Castleman, Pearson.

4th Year 7th Semester									
Sl. No.	**Broad Category**	**Category**	**Paper Code**	**Subject**	**L**	**T**	**P**	**Total**	**Credit Points**
A. Theory									
3	ENGG	Major	CB701A	Big Data Analytics	3	0	0	3	3
			CB701B	Deep Learning and Neural Networks					
			CB701C	Cybersecurity and Ethical Hacking					
			CB701D	Pattern Recognition					
3	ENGG	Major	CB702A	Natural Language Processing	3	0	0	3	3
			CB702B	Introduction to Blockchain					
			CB702C	Quantum Computing Fundamentals					
			CB702D	Introduction to Game Development					
3	ENGG	Major	CB703A	Mobile Computing	3	0	0	3	3
			CB703B	Real-Time Operating Systems					
			CB703C	Introduction to Data Science					
			CB703D	Cloud Computing					
3	ENGG	Minor	HU(CB)701A	Psychology for Engineers	3	0	0	3	3
			HU(CB)701B	Digital Transformation in Business					
			EE(CB)701C	Green Computing					
			IT(CB)701D	Web Technology					
B. Practical									
3	ENGG	Major	CB791A	Big Data Analytics Lab	0	0	3	3	1.5
			CB791B	Deep Learning and Neural Networks Lab					
			CB791C	Cybersecurity and Ethical Hacking Lab					
			CB791D	Pattern Recognition Lab					
3	ENGG	Major	CB792A	Natural Language Processing Lab	0	0	3	3	1.5
			CB792B	Introduction to Blockchain Lab					
			CB792C	Quantum Computing Fundamentals Lab					
			CB792D	Introduction to Game Development Lab					
4	PRJ	PRJ	CB781	Project-III	0	0	12	12	6
Total of Theory, Practical								**27**	**21**

Course Name: Big Data Analytics

Course Code: CB701A

Contact (Periods/Week): 3 periods

Total Contact Hours: 36 Hours

Credits: 3

Course Objective(s):

- Understand the Big Data platform and common use cases.
- Learn Hadoop architecture, HDFS, and the MapReduce programming model.
- Work with Hadoop ecosystem tools including Pig, Hive, and HBase.
- Apply big data analytics techniques on structured and unstructured data.
- Integrate R programming and machine learning methods for data analysis and visualization.

Course Outcomes (COs):

After successful completion of the course, students will be able to:

CO1: Identify Big Data fundamentals and their business applications.

CO2: Explain the architecture and components of the Hadoop ecosystem.

CO3: Access and process data using HDFS, Flume, Sqoop, and related tools.

CO4: Develop and manage MapReduce jobs for large-scale data processing.

CO5: Build and analyze data pipelines using Pig, Hive, HBase, and R for big data analytics.

CO-PO-PSO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	2	2	2	0	0	0	0	0	0	2	2	2
CO2	3	3	3	2	2	0	2	0	0	0	0	3	3	2
CO3	3	3	3	3	3	2	0	0	2	0	0	3	3	3
CO4	3	3	3	3	3	2	0	0	2	0	0	3	3	3
CO5	3	3	3	3	3	3	3	2	2	0	0	3	3	3

Course Contents

1. **Module I: Introduction to Big Data and Hadoop (6L)**
    - Types of Digital Data, Introduction to Big Data and its Analytics
    - History and architecture of Hadoop, Hadoop Streaming
    - Analyzing Data using Unix tools and Hadoop

- Overview of Hadoop Ecosystem
- IBM Big Data Strategy, Infosphere BigInsights, BigSheets

2. **Module II: Hadoop Distributed File System (HDFS) (6L)**
    - Design and Concepts of HDFS
    - HDFS Command Line Interface and APIs
    - Data flow in HDFS, Data ingestion using Flume and Sqoop
    - Hadoop archives, Compression and Serialization, Avro
    - File-Based Data Structures
3. **Module III: MapReduce Programming Model (6L)**
    - Anatomy of a MapReduce Job
    - Failures, Job Scheduling, Shuffle and Sort
    - Task Execution and Job Monitoring
    - MapReduce Types, Input/Output Formats and Features
4. **Module IV: Hadoop Ecosystem (9L)**
    - Pig: Execution Modes, Pig Latin, Grunt Shell, UDFs, Operators
    - Hive: Hive Shell, Services, Metastore, HiveQL, UDFs, Table Operations
    - HBase: Architecture, Clients, HBase vs RDBMS
    - Big SQL: Introduction and Features
5. **Module V: Data Analytics with R (9L)**
    - Machine Learning Basics: Supervised, Unsupervised Learning, Collaborative Filtering
    - Big Data Analytics with BigR
    - R Integration with Hadoop: RHadoop ecosystem
    - Building Predictive Models and Data Visualization using R

Text Books:

- Tom White – *Hadoop: The Definitive Guide*, 3rd Edition, O'Reilly Media, 2012
- Seema Acharya and Subhasini Chellappan – *Big Data Analytics*, Wiley, 2015

Reference Books:

- Michael Berthold, David J. Hand – *Intelligent Data Analysis*, Springer, 2007
- Jay Liebowitz – *Big Data and Business Analytics*, CRC Press, 2013
- Tom Plunkett, Mark Hornick – *Using R to Unlock the Value of Big Data*, McGraw-Hill, 2013
- Anand Rajaraman, Jeffrey D. Ullman – *Mining of Massive Datasets*, Cambridge University Press, 2012
- Bill Franks – *Taming the Big Data Tidal Wave*, John Wiley & Sons, 2012
- Glen J. Myatt – *Making Sense of Data*, John Wiley & Sons, 2007
- Pete Warden – *Big Data Glossary*, O'Reilly, 2011
- Michael Minelli et al. – *Big Data, Big Analytics*, Wiley, 2013
- Arvind Sathi – *Big Data Analytics: Disruptive Technologies*, MC Press, 2012
- Paul Zikopoulos et al. – *Harness the Power of Big Data: The IBM Big Data Platform*, McGraw Hill, 2012

Course Name: Deep Learning and Neural Networks

Course Code: CB701B

Contact(Periods/Weeks): 3 Periods

Total Contact Hours: 36 hrs

Credits: 3

Course Objective(s):

The objective of the course is to make the students able to:

- Introduce the theoretical foundations of neural networks and deep learning.
- Explain optimization methods, loss functions, and regularization strategies.
- Develop and train deep learning architectures such as CNNs, RNNs, and advanced models.
- Apply deep learning methods to real-world problems in computer vision, speech, and NLP.
- Explore recent trends in generative models, transformers, and ethical AI.

Course Outcomes (COs):

After successful completion of the course, students will be able to:

CO1: Understand the mathematical foundations of neural networks and deep learning.

CO2: Implement basic neural network models including perceptrons, feedforward, and multilayer architectures.

CO3: Apply optimization, backpropagation, and regularization techniques to train deep models effectively.

CO4: Analyze and design convolutional neural networks (CNNs), recurrent neural networks (RNNs), and modern architectures.

CO5: Apply deep learning models to solve real-world problems in image, speech, and natural language processing.

CO–PO–PSO Mapping

	PO1	PO2	PO3	PO4	PO5	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	–	–	2	–	–	–	–	3	–	–
CO2	–	3	3	–	2	–	–	–	–	3	2	–
CO3	–	3	–	3	2	–	–	–	–	2	3	2
CO4	–	2	–	3	3	–	–	2	–	3	3	–
CO5	–	–	3	2	3	2	2	–	3	–	3	3

Course Contents

Unit I: Introduction and Foundations

- Overview of AI, Machine Learning, and Deep Learning.
- Biological vs. Artificial Neural Networks.
- Basics of linear algebra, probability, and optimization for neural networks.
- Perceptrons and Feedforward Neural Networks.

Unit II: Training Neural Networks

- Gradient Descent and Backpropagation.
- Loss functions: MSE, Cross-Entropy.
- Regularization techniques: L1, L2, Dropout, Batch Normalization.
- Optimization algorithms: SGD, Momentum, RMSProp, Adam.

Unit III: Deep Architectures

- Multilayer Perceptrons (MLP).
- Convolutional Neural Networks (CNNs): convolution, pooling, architectures (LeNet, AlexNet, VGG, ResNet).
- Recurrent Neural Networks (RNNs): vanilla RNNs, vanishing/exploding gradients.
- Advanced RNN variants: LSTM, GRU.

Unit IV: Generative and Advanced Models

- Autoencoders and Variational Autoencoders (VAE).
- Generative Adversarial Networks (GANs).
- Transfer Learning and Fine-Tuning.
- Attention Mechanisms and Transformers (BERT, GPT).

Unit V: Applications and Trends

- Deep Learning for Image Recognition, Object Detection, and Computer Vision.
- Deep Learning for Speech Processing and Natural Language Understanding.
- Reinforcement Learning with Deep Networks.
- Ethical AI, interpretability, and future trends in Deep Learning.

Text Books:

- Ian Goodfellow, Yoshua Bengio, Aaron Courville – *Deep Learning*, MIT Press, 2016.
- Michael Nielsen – *Neural Networks and Deep Learning*, Determination Press, 2015.
- Franc¸ois Chollet – *Deep Learning with Python*, Manning, 2021.

Reference Books:

- Christopher Bishop – *Pattern Recognition and Machine Learning*, Springer, 2006.
- Charu Aggarwal – *Neural Networks and Deep Learning*, Springer, 2018.
- Trevor Hastie, Robert Tibshirani, Jerome Friedman – *The Elements of Statistical Learning*, Springer, 2009.
- Richard S. Sutton, Andrew G. Barto – *Reinforcement Learning: An Introduction*, MIT Press, 2018.

Course Name: Cybersecurity And Ethical Hacking

Course Code: CB701C

Contact (Periods/Week): 3 periods

Total Contact Hours: 36 Hours

Credits: 3

Prerequisites:

Basic knowledge of computer networks, operating systems, and programming fundamentals.

Course Objective(s):

- Understand the fundamentals of cybersecurity, threat models, and ethical hacking.
- Gain hands-on experience with Kali Linux, VPN, IP, MAC spoofing, and anonymization tools.
- Learn footprinting, open-source intelligence gathering, and social engineering techniques.
- Perform network mapping, vulnerability analysis, malware investigation, and DoS prevention.
- Explore website penetration testing, OWASP Top 10 vulnerabilities, and web application tools.

Course Outcomes (COs):

After successful completion of the course, students will be able to:

CO1: Explain cybersecurity concepts, threat models, CIA triad, and cyber laws.

CO2: Operate Kali Linux and tools for secure browsing, anonymization, and networking.

CO3: Perform reconnaissance using open-source intelligence and defend against phishing attacks.

CO4: Conduct network scanning, vulnerability analysis, and malware inspection.

CO5: Execute web-based attacks and defense mechanisms through penetration testing tools.

CO-PO-PSO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	2	2	2	0	0	0	0	0	2	1	0	1
CO2	3	3	3	2	3	0	0	0	2	0	3	1	0	1
CO3	3	3	3	3	3	0	0	0	2	0	3	1	0	1
CO4	3	3	3	3	3	0	0	0	2	0	3	1	0	1
CO5	3	3	3	3	3	0	0	0	2	0	3	1	0	1

Course Contents

1. **Module I: Cybersecurity Essentials (5L)**
   - Introduction to Cybersecurity
   - OSI and TCP/IP models
   - Cyberattack types, threat categories, attack vectors
   - Risk assessment, CIA triad, stages and types of hacking
   - Cyber laws, standards, and ethical hacking overview
2. **Module II: Kali Linux, VPNs, and Anonymity (5L)**
   - Kali Linux installation and interface overview
   - Basic Linux commands and file operations
   - Deep Web, Dark Web, Tor browser
   - IP/MAC address concepts, VPN and proxy usage
   - Spoofing and anonymization techniques
3. **Module III: Footprinting and Social Engineering (7L)**
   - Information gathering: active and passive techniques
   - Footprinting outcomes and OSINT tools
   - Tools: Maltego, Recon-ng, Hunter.io
   - Art of Googling and subdomain enumeration
   - Social engineering: phishing types and countermeasures
4. **Module IV: Network Vulnerabilities and Malware (7L)**
   - TCP, UDP basics and common port numbers
   - Network mapping and scanning
   - Tools: Nessus, Nikto, Lynis, Metasploit, Searchsploit
   - Android hacking, malware analysis, code review
   - DoS and DDoS: symptoms, tools (Hammer, DDoSTor), prevention
5. **Module V: Web Attacks and Penetration Testing (6L)**
   - Website attack types, penetration testing methods
   - OWASP TOP 10, web application scanners (ZAP, BurpSuite)
   - Cross-site scripting (XSS), SQL injection, SQLMap
   - Sniffing and network security
   - Project / Hackathon: Live penetration testing or vulnerability analysis
6. **Module VI: Professional Communication and Ethics (6L)**
   - Communicative English in a professional environment
   - Information technology awareness
   - Professional and cybersecurity ethics

Text Books:

- Class Handouts and Instructor Notes (aligned with hands-on tools and open-source materials)

Reference Books:

- Kali Linux Revealed, Offensive Security
- OWASP Top 10 Web Application Security Risks
- Dafydd Stuttard and Marcus Pinto, *The Web Application Hacker's Handbook*
- Tools: Metasploit, BurpSuite, SQLMap, Nessus, Maltego, ZAP, Recon-ng, Wireshark
- Cybersecurity and Infrastructure Security Agency (CISA)
- MITRE ATT&CK Framework

Course Name: Pattern Recognition

Course Code: CB701D

Contact(Periods/Weeks): 3 Periods

Total Contact Hours: 36 hrs

Credits: 3

Course Objective(s):

The objective of this course is to introduce the fundamental concepts and methodologies of pattern recognition. Students will gain knowledge of classification, clustering, feature extraction, and dimensionality reduction techniques. The course also emphasizes the application of pattern recognition algorithms in domains such as image analysis, speech recognition, and data mining.

Course Outcomes (COs):

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

CO1: Understand the theoretical foundations of pattern recognition and decision-making strategies.

CO2: Apply statistical, structural, and neural methods for classification and clustering problems.

CO3: Implement feature extraction, feature selection, and dimensionality reduction techniques.

CO4: Evaluate and compare different pattern recognition algorithms with respect to accuracy, robustness, and efficiency.

CO5: Design and apply pattern recognition solutions to real-world problems in image processing, speech recognition, and data mining.

CO–PO–PSO Mapping

	PO1	PO2	PO3	PO4	PO5	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	–	–	2	–	–	–	–	3	–	–
CO2	–	3	3	–	2	–	–	–	–	2	3	–
CO3	–	3	–	3	2	–	–	–	–	2	2	2
CO4	–	2	–	2	–	–	–	2	–	3	3	–
CO5	–	–	3	2	3	2	2	–	3	–	3	3

Course Contents

Unit I: Introduction to Pattern Recognition

- Definitions and applications of pattern recognition.
- Learning paradigms: supervised, unsupervised, semi-supervised, and reinforcement learning.
- Decision boundaries, distance measures, and similarity metrics.

Unit II: Statistical Pattern Recognition

- Bayes' decision theory, discriminant functions.

- Maximum likelihood estimation, Bayesian estimation.
- Parametric vs. non-parametric techniques: k-NN, Parzen windows, decision trees.

Unit III: Feature Extraction and Dimensionality Reduction

- Feature selection methods: branch and bound, sequential selection.
- Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA).
- Independent Component Analysis (ICA) and manifold learning.

Unit IV: Clustering Techniques

- Similarity measures and clustering validity.
- Partitioning methods: k-means, k-medoids.
- Hierarchical clustering: agglomerative and divisive methods.
- Density-based clustering: DBSCAN, OPTICS.
- Model-based clustering: Gaussian Mixture Models, Expectation-Maximization.

Unit V: Advances and Applications in Pattern Recognition

- Structural pattern recognition and syntactic approaches.
- Neural networks, SVMs, ensemble methods.
- Fuzzy and neuro-fuzzy approaches.
- Applications: image recognition, speech recognition, text mining, biometrics.

Assessment Method

- Internal Evaluation: Quizzes, Assignments, Mini-Projects (30%)
- Mid-Semester Examination (20%)
- End-Semester Examination (50%)

Learning Resources

Textbooks / References:

1. Christopher M. Bishop, *Pattern Recognition and Machine Learning*, Springer, 2006.
2. Trevor Hastie, Robert Tibshirani, Jerome Friedman, *The Elements of Statistical Learning*, Springer, 2009.
3. Ulisses Braga-Neto, *Fundamentals of Pattern Recognition and Machine Learning*, Springer, 2020.
4. Papoulis & Pillai, *Probability, Random Variables and Stochastic Processes*, 4th Edition, Tata McGraw Hill.
5. Gilbert Strang, *Linear Algebra and Its Applications*, Thomson Books.
6. Jiawei Han, Micheline Kamber, *Data Mining: Concepts and Techniques*, Morgan Kaufmann Publishers.
7. A. K. Jain & R. C. Dubes, *Algorithms for Clustering Data*, Prentice Hall, 1988.

Course Name: Natural Language Processing

Course Code: CB702A

Contact (PeriodsWeeks): 3 Periods

Total Contact Hours: 36 hrs

Credits: 3

Course Objective(s):

The objective of this course is to introduce students to the fundamental concepts, models, and applications of Natural Language Processing (NLP). Students will learn linguistic foundations, text processing techniques, and statistical as well as neural approaches for solving real-world NLP tasks.

Course Outcomes (COs):

After successful completion of the course, students will be able to:

CO1: Understand the basics of text processing, morphology, and syntax in natural languages.

CO2: Apply probabilistic and statistical models such as n-gram language models, HMMs, and CRFs in NLP tasks.

CO3: Implement and evaluate parsing techniques for constituency and dependency structures.

CO4: Analyze semantic representation methods including WordNet, distributional semantics, and topic models.

CO5: Design and develop NLP applications such as sentiment analysis, machine translation, text summarization, and question answering.

CO–PO Mapping

CO–PO–PSO Mapping

	PO1	PO2	PO3	PO4	PO5	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	–	–	2	–	–	–	–	3	–	–
CO2	–	3	3	–	2	–	–	–	–	2	3	–
CO3	–	3	–	3	2	–	–	–	–	2	2	2
CO4	–	2	–	2	–	–	–	2	–	3	3	–
CO5	–	–	3	2	3	2	2	–	3	–	3	3

Course Contents

Unit I: Introduction & Text Processing

- Introduction to NLP, challenges in natural language understanding.
- Text pre-processing: tokenization, stemming, lemmatization, stop-word removal.
- Spelling correction and edit distance.

- Language modeling: n-gram models, smoothing techniques.

Unit II: Morphology, POS Tagging, and Sequential Models

- Morphology fundamentals; finite-state methods for morphology.
- POS tagging: Hidden Markov Models, Maximum Entropy models.
- Sequence labeling: CRFs and applications.

Unit III: Parsing and Syntax

- Constituency parsing: top-down and bottom-up approaches, probabilistic CFGs.
- Dependency parsing: transition-based and graph-based methods.
- Handling ambiguity in parsing.

Unit IV: Semantics and Meaning Representation

- Lexical semantics: WordNet, distributional semantics, semantic similarity.
- Word Sense Disambiguation (WSD) and semantic role labeling.
- Topic modeling: LDA and applications.

Unit V: Applications of NLP

- Sentiment analysis and opinion mining.
- Machine Translation: statistical and neural approaches.
- Text summarization techniques.
- Information extraction: Named Entity Recognition, relation extraction.
- Question Answering and dialogue systems.

Assessment Method

- Internal Evaluation: Quizzes, Assignments, Case Studies (30%)
- Mid-Semester Examination (20%)
- End-Semester Examination (50%)

Learning Resources

Textbooks / References:

1. Daniel Jurafsky & James H. Martin, *Speech and Language Processing*, 3rd Edition (Draft), Pearson, 2023.
2. Christopher Manning & Hinrich Schütze, *Foundations of Statistical Natural Language Processing*, MIT Press, 1999.
3. James Allen, *Natural Language Understanding*, 2nd Edition, Pearson, 1995.
4. Jacob Eisenstein, *Introduction to Natural Language Processing*, MIT Press, 2019.
5. Palash Goyal, Sumit Pandey, Karan Jain, *Deep Learning for Natural Language Processing*, Apress, 2018.

Course Name: Introduction To Blockchain

Course Code: CB702B

Contact (Periods/Week): 3 periods

Total Contact Hours: 36 Hours

Credits: 3

Prerequisites:

Basic knowledge of computer networks, cryptography, and programming.

Course Objective(s):

- Introduce the foundational concepts and evolution of blockchain technology.
- Explore decentralized systems, smart contracts, and distributed ecosystems.
- Understand blockchain architecture, cryptocurrencies, and Bitcoin mechanisms.
- Examine blockchain platforms and their applications across industries.
- Enable students to evaluate, design, and apply blockchain-based solutions.

Course Outcomes (COs):

After successful completion of the course, students will be able to:

CO1: Describe blockchain fundamentals, types, evolution, and the concept of decentralization.

CO2: Analyze blockchain architecture, cryptocurrency systems, and Bitcoin mechanisms.

CO3: Apply cryptographic and network concepts like hashing, double spending, wallets, and APIs.

CO4: Evaluate and compare popular blockchain platforms and their real-world applications.

CO5: Design basic blockchain-based solutions using decentralized platforms and evaluate their potential impact in real-world applications.

CO-PO-PSO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	2	2	1	0	0	0	0	0	2	2	2	2
CO2	3	3	3	2	3	0	0	0	2	0	3	2	2	2
CO3	3	3	3	3	3	0	0	0	2	0	3	2	2	1
CO4	3	3	3	3	3	2	0	0	2	0	3	2	2	2
CO5	3	3	3	3	3	2	2	2	3	0	3	3	3	3

Course Contents

1. **Module I: Discovering Blockchain & Decentralization (9L)**
   - Introduction to blockchain and distributed systems
   - Growth and history of blockchain and Bitcoin
   - Types of blockchain: Public, Private, Consortium
   - Methods and routes of decentralization
   - Smart contracts and decentralized applications (DApps)
   - Decentralized Autonomous Organizations (DAOs)
2. **Module II: Blockchain Architecture & Cryptocurrencies (9L)**
   - Blockchain architecture, versions, and variants
   - Blockchain vs traditional/shared databases
   - Real-life blockchain use cases across industries
   - Introduction to cryptocurrencies: definition, types, and applications
   - Bitcoin fundamentals: keys, addresses, transactions
   - Blockchain mining and limitations of Bitcoin
   - Overview of alternative coins (Altcoins)
3. **Module III: Cryptographic Concepts & Bitcoin Network (9L)**
   - Double spending problem
   - Hashing and proof-of-work mechanisms
   - Bitcoin network architecture
   - Bitcoin wallets and payment flow
   - APIs and clients for Bitcoin development
   - Innovations and improvements in Bitcoin ecosystem
4. **Module IV: Blockchain Platforms and Applications (9L)**
   - Overview of platforms: Ethereum, Hyperledger, IOTA, EOS, Multichain, BigchainDB
   - Comparative analysis: Ethereum vs Bitcoin
   - Advantages and disadvantages of various blockchain platforms
   - Designing a new blockchain and DApp
   - Potential for disruption in supply chain, healthcare, finance, and digital identity
   - Emerging trends and future directions

Text Books:

- Blockchain Basics: A Non-Technical Introduction in 25 Steps – Daniel Drescher, Apress.

Reference Books:

- Mastering Bitcoin – Andreas M. Antonopoulos, O'Reilly.
- Mastering Blockchain – Imran Bashir, Packt Publishing.
- Ethereum Developer Docs
- Hyperledger Frameworks
- Solidity Language Documentation
- Bitcoin Whitepaper – Satoshi Nakamoto
- IOTA Foundation

Course Name: Introduction to Game Development

Course Code: CB702D

Contact (Periods/Week): 3 periods

Total Contact Hours: 36 Hours

Credits: 3

Prerequisites:

Basic knowledge of programming like OOPs.

Course Description

This course introduces students to the fundamentals of game development. Topics include game design principles, 2D and 3D graphics basics, game physics, animation, input handling, audio, game engines (Unity/Unreal overview), scripting, asset pipelines, optimization, and an introduction to multiplayer and publishing considerations. The course combines lectures with hands-on lab work where students will build small games and game prototypes.

Course Objectives (COs)

- Understand core concepts of game design and software architecture for games.
- Implement 2D and basic 3D graphics, camera, and scene management.
- Apply physics, collision detection, and animation techniques in games.
- Use a modern game engine (e.g., Unity or Unreal) to prototype interactive games.
- Optimize game performance, manage assets and implement basic audio & input systems.
- Work in teams to design, develop, and present a small playable game prototype.

Course Outcomes (COs) - Measurable

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

CO1: Explain game design principles, game loop architecture, and common design patterns used in games. CO2:

Develop basic 2D games and implement simple 3D rendering pipelines using engine APIs or libraries. CO3:

Implement physics-based motion, collision detection/resolution, and skeletal or sprite-based animation. CO4:

Build and deploy a playable game prototype using a game engine and scripting.

CO5: Profile and optimize a game's performance (rendering, memory, and CPU) and handle asset management.

CO-PO-PSO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3		2	1	2	0	0	0	0	0	2	2	1	2
CO2	2	2	3	2	2	0	0	0	2	0	3	2	2	2
CO3	1	3	3	3	3	2	0	0	2	0	3	2	2	2
CO4	2	2	3	3	3	2	0	0	2	0	3	2	2	2
CO5	3	3	3	3	3	2	1	0	2	0	3	3	3	3

Course Content (36 Lectures)

Week	Topics (Lectures)
1	Introduction to games and genres; history of games; overview of the game development pipeline; roles in game teams; the game loop; basic architecture. (3L)
2	Game design fundamentals: mechanics, dynamics, aesthetics (MDA); level design basics; prototyping and iteration; introduction to Unity/Unreal interfaces. (3L)
3	2D graphics: coordinate systems, sprites, tilemaps, rendering order, simple shaders; camera systems for 2D. (3L)
4	3D graphics basics: 3D coordinate spaces, transforms, matrices, cameras, projection (orthographic vs perspective), scene graphs. (3L)
5	Textures, materials, lighting models (Phong, Blinn-Phong), introduction to PBR concepts. (3L)
6	Input systems: keyboard, mouse, touch, controllers; UI basics and HUD design. (3L)
7	Game physics I: kinematics, forces, simple integrators, gravity, discrete time stepping. (3L)
8	Game physics II: collision detection (AABB, sphere, SAT), collision resolution, rigid body basics. (3L)
9	Animation: sprite animation, skeletal animation, blending, inverse kinematics overview. (3L)
10	Audio in games: sound effects, music, spatial audio basics, mixing and triggers. (2L)
11	Scripting and gameplay programming patterns; event systems; state machines; AI basics (FSM, simple pathfinding). (3L)
12	Asset workflows and pipeline (art, audio, levels); importing/exporting assets; version control best practices for game projects. (3L)
13	Optimization techniques: draw calls, batching, culling, LOD, memory management, profiler usage. (3L)
14	Introduction to multiplayer: client-server vs peer-to-peer, latency, synchronization basics; security and cheating considerations. (2L)
15	Game production considerations: publishing platforms, monetization models, testing and QA, accessibility and ethics. (2L)

Recommended Text Books

1. **”Fundamentals of Game Design”**, 3rd Edition, Ernest Adams and Andrew Rollings, Peachpit Press.
2. **”Game Engine Architecture”**, 3rd Edition, Jason Gregory, CRC Press.
3. **”Unity in Action: Multiplatform Game Development in C# with Unity”**, 2nd Edition, Joseph Hocking, Manning Publications.

Reference Books

1. **”Real-Time Rendering”**, 4th Edition, Tomas Akenine-M0̆0f6ller, Eric Haines, Naty Hoffman, CRC Press.
2. **”Game Programming Patterns”**, Robert Nystrom, Genever Benning (online/book).

Course Name: Mobile Computing

Course Code: CB703A

Contact (Periods/Week): 3 periods

Total Contact Hours: 36 Hours

Credits: 3

Prerequisites:

Basic knowledge of networking, communication systems, and computer architecture.

Course Objective(s):

- Understand the evolving trends in mobile computing and their business impact.
- Learn mobile design principles relevant to wireless and mobile applications.
- Study the architecture and functioning of wireless networks.
- Evaluate next-generation communication systems, ad hoc networks, and mobile frameworks.
- Explore D2D, cognitive radio, and real-time wireless sensor network applications.

Course Outcomes (COs):

After successful completion of the course, students will be able to:

CO1: Recall the fundamental concepts of mobile communication and networking.

CO2: Understand and interpret the mobile and transport layers in wireless systems.

CO3: Analyze next-generation mobile communication systems and technologies.

CO4: Examine wireless transmission principles and Mobile Ad-hoc Network routing.

CO5: Evaluate real-time sensor networks, cognitive radio, and device-to-device communication in 5G.

CO-PO-PSO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	2	2	1	2	0	0	0	0	0	2	2	1	2
CO2	3	3	3	2	2	0	0	0	2	0	3	2	2	2
CO3	3	3	3	3	3	0	0	0	2	0	3	2	2	2
CO4	3	3	3	3	3	0	0	0	2	0	3	2	2	2
CO5	3	3	3	3	3	0	0	0	2	0	3	3	3	3

Course Contents

1. **Module I: Mobile Infrastructure, Location, and Handoff Management (6L)**
   - Overview of mobile and wireless infrastructure
   - Cellular architecture: design goals, frequency reuse, radio resource management
   - Propagation models, interference, cell splitting
   - Mobile generations: 1G to 5G
   - Location management (HLR, VLR), mobility models (Random walk, Fluid flow, Markovian)
   - Location update strategies (static/dynamic), terminal paging
   - Mobile IP, handoff types and metrics (soft, hard, horizontal, vertical)
2. **Module II: Wireless Transmission and Ad-hoc Networks (6L)**
   - Narrowband vs wideband systems, Spread Spectrum, Frequency Hopping
   - Basics of MIMO and MIMO-OFDM
   - Multiple access control: FDMA, TDMA, CDMA, SDMA
   - WLAN, WPAN (Bluetooth, ZigBee)
   - Mobile Ad-hoc Networks (MANETs): characteristics, applications, routing, coverage challenges
3. **Module III: Wireless Sensor Networks and Cognitive Radio (6L)**
   - WSN architecture, sensing/communication range, coverage/connectivity
   - Energy-efficient routing (e.g., LEACH), clustering and data aggregation
   - Cognitive Radio Networks: spectrum sensing (fixed/dynamic), spectrum sharing
   - Interoperability and co-existence, applications in smart systems
4. **Module IV: Device-to-Device (D2D) and 5G Communication (6L)**
   - Introduction to D2D communications
   - 5G architecture: requirements, millimeter wave communication
   - Resource management, power control, and mode selection in D2D
   - Application domains for D2D (e.g., public safety, IoT integration)

Text Books:

- Jochen Schiller, *Mobile Communications*, Pearson Education, 2nd Edition, 2008
- Andrea Goldsmith, *Wireless Communications*, Cambridge University Press, 2005

Reference Books:

- Theodore Rappaport, *Wireless Communications: Principles and Practice*, Pearson
- Ezio Biglieri, *Wireless Communications MIMO*, Cambridge University Press
- Ivan Stojmenovic, *Handbook of Wireless Networking and Mobile Computing*, Wiley
- James Cowling, *Dynamic Location Management in Heterogeneous Cellular Networks*
- Gordon L. Stüber, *Principles of Mobile Communication*, Springer
- Lingyang Song et al., *Wireless Device-to-Device Communications and Networks*, Cambridge
- Ezio Biglieri et al., *Principles of Cognitive Radio*, Cambridge
- Edgar H. Callaway Jr., *Wireless Sensor Networks: Architectures and Protocols*, CRC Press

Web References:

- https://www.nsnam.org/docs/manual/html/index.html
- http://people.csail.mit.edu/cowling/hons/jcowling-dynamic-Nov04.pdf
- https://www.cse.wustl.edu/~jain/cse574-06/ftp/cellular_location.pdf

Course Name: Introduction To Data Science

Course Code: CB703C

Contact (Periods/Week): 3 periods

Total Contact Hours: 36 Hours

Credits: 3

Prerequisites:

Basic programming in Python, knowledge of statistics, and linear algebra fundamentals.

Course Objective(s):

- Introduce the basic concepts and tools of data science using Python.
- Explore statistical analysis and inference techniques for data-driven decision-making.
- Apply supervised learning and regression techniques to real-world datasets.
- Understand unsupervised learning, clustering methods, and similarity measures.
- Analyze and visualize networks and graph-based data structures.

Course Outcomes (COs):

After successful completion of the course, students will be able to:

CO1: Describe the fundamentals of data science and essential Python tools.

CO2: Perform exploratory data analysis (EDA) using descriptive and inferential statistics.

CO3: Understand and apply supervised learning algorithms such as SVM and Random Forest.

CO4: Build and evaluate regression models for predictive analytics.

CO5: Explore and analyze network data using centrality, community detection, and PageRank.

CO-PO-PSO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2	PSO3
CO1	3	3	2	2	2	0	0	0	0	0	0	2	2	2
CO2	3	3	3	2	2	0	0	0	2	0	0	3	2	2
CO3	3	3	3	3	3	0	0	0	2	0	0	3	3	3
CO4	3	3	3	3	3	0	0	0	2	0	0	3	3	3
CO5	3	3	3	3	3	2	0	0	2	0	0	3	3	3

Course Contents

1. **Module I: Introduction and Data Manipulation with Python (6L)**
   - Introduction to Data Science
   - Toolboxes: Python and essential libraries (NumPy, Pandas, Matplotlib, Seaborn)
   - Working with IDEs (Jupyter, Spyder, VS Code)
   - Data operations: reading, selecting, filtering, manipulating, sorting, grouping, ranking, and plotting
2. **Module II: Descriptive Statistics and Inference (6L)**
   - Descriptive statistics, data summarization, distributions, and asymmetry
   - Data preparation techniques
   - Statistical inference: frequency approach, variability of estimates
   - Hypothesis testing: confidence intervals and p-values
3. **Module III: Supervised Learning (8L)**
   - Learning curves and dataset splitting (training-validation-test)
   - General learning model structure
   - Classification models: Support Vector Machines (SVM), Random Forest
   - Case studies and practical applications
4. **Module IV: Regression and Unsupervised Learning (8L)**
   - Regression analysis: linear, multiple, polynomial, and sparse models
   - Clustering: k-means, hierarchical clustering
   - Similarity and distance measures
   - Clustering quality measures
   - Case study in unsupervised learning
5. **Module V: Network Analysis and Social Graphs (8L)**
   - Graph structures and fundamentals of network analysis
   - Centrality measures and graph visualization
   - PageRank algorithm and Ego-Networks
   - Community detection techniques
   - Applications in social networks

Text Books:

- Laura Igual & Santi Segu´ı – *Introduction to Data Science: A Python Approach to Concepts, Techniques and Applications*, Springer, ISBN: 978-3-319-50016-4
- David Taieb – *Data Analysis with Python: A Modern Approach*, Packt Publishing, ISBN: 9781789950069
- Armando Fandango – *Python Data Analysis*, 2nd Edition, Packt Publishing, ISBN: 9781787127487

Course Name: Cloud Computing

Course Code: CB703D

Contact (Periods/Week): 3 periods

Total Contact Hours: 36 Hours

Credits: 3

Course Objective(s):

- Comprehend foundational concepts, benefits, and challenges of cloud computing.
- Analyze cloud service models (IaaS, PaaS, SaaS) and deployment strategies.
- Evaluate virtualization technologies and cloud infrastructure components.
- Assess security, privacy, and compliance issues in cloud environments.
- Develop practical cloud applications using industry-standard platforms.

Course Outcomes (COs):

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

CO1: Understand the fundamentals and architecture of cloud computing.

CO2: Explain the key cloud delivery models and deployment strategies.

CO3: Demonstrate knowledge of virtualization and cloud infrastructure components.

CO4: Analyze the security and privacy issues in cloud-based environments.

CO5: Develop cloud-based applications using real-world platforms.

CO-PO Mapping

CO-PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	3	2	0	0	0	0	0	0	0	0	0	0
CO2	3	3	2	0	0	0	0	0	0	0	0	0
CO3	2	2	3	2	0	0	0	0	0	0	0	0
CO4	2	2	3	2	0	0	2	0	0	0	0	0
CO5	3	2	3	2	2	0	0	2	2	0	2	0

Course Contents

1. **Module I: Introduction to Cloud Computing (4L)**
   - Definition, Benefits, Challenges, and Risks
   - Characteristics of Cloud Computing
   - Evolution of Cloud Computing
   - Role of Cloud in Modern IT

2. **Module II: Cloud Computing Architecture (5L)**
   - Composability, Infrastructure, Platforms, Virtual Appliances
   - Communication Protocols, Applications, Connecting to the Cloud by Clients
   - Components of Cloud Computing Architecture
   - Service Models (IaaS, PaaS, SaaS)
   - Deployment Models (Public, Private, Hybrid, Community)
   - Case Studies

3. **Module III: Virtualization in Cloud (8L)**
   - Basics of Virtualization
   - Types of Virtualizations (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 load balancing
   - Hypervisors
   - Virtual Machine (VM) Provisioning and Management
   - Containerization using Docker and Kubernetes

4. **Module IV: Cloud Service Providers and Tools (7L)**
   - Overview of AWS, Microsoft Azure, Google Cloud Platform
   - Services like EC2, S3, Lambda, App Engine
   - Deployment Examples

5. **Module V: Security in Cloud Computing (6L)**
   - Cloud Security Challenges
   - Data Security
   - Identity and Access Management
   - Standards and Compliance
   - Cloud Security Alliance (CSA) guidelines

6. **Module VI: Emerging Trends and Applications (6L)**
   - Serverless Computing
   - Edge and Fog Computing
   - Cloud-native Applications
   - Industry Use Cases
   - Future of Cloud Computing

Text Books:

- Rajkumar Buyya, Christian Vecchiola, and S. Thamarai Selvi – *Mastering Cloud Computing*, McGraw Hill Education
- Toby Velte, Anthony Velte, Robert Elsenpeter – *Cloud Computing: A Practical Approach*, McGraw Hill

Reference Books:

- Arshdeep Bahga, Vijay Madisetti – *Cloud Computing: A Hands-On Approach*
- Thomas Erl – *Cloud Computing: Concepts, Technology & Architecture*, Pearson
- George Reese – *Cloud Application Architectures*, O'Reilly

Course Name: Psychology for Engineers

Course Code: HU(CB)701A

Contact (Periods/Week): 3 periods

Total Contact Hours: 48:00 hrs

Credits: 3

Prerequisites: None

Course Objective(s):

- To familiarize engineering students with foundational psychological principles.
- To develop awareness about human cognition, behavior, perception, learning, and decision-making.
- To enhance skills for effective communication, design thinking, collaboration, and leadership from a psychological perspective.

Course Outcomes (COs):

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

CO1: Understand core concepts of psychology including attention, perception, learning, and memory.

CO2: Analyze human behavior in the context of engineering environments (teams, workplaces, systems).

CO3: Apply psychological theories to solve human-centered problems in engineering design.

CO4: Demonstrate effective communication, leadership, and emotional intelligence using psychological insights.

CO5: Evaluate user behavior and cognitive ergonomics in the development of safe, efficient systems.

CO-PO Mapping

CO-PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	2	1	0	0	0	0	0	0	0	0	0	2	0
CO2	1	2	0	0	0	2	0	0	3	3	0	0	2
CO3	3	2	3	0	2	1	0	0	3	2	0	3	0
CO4	0	0	0	0	0	0	0	2	3	2	1	0	3
CO5	2	3	2	2	2	0	2	2	2	2	0	2	0

Course Contents

1. **Module 1: Introduction to Psychology: Historical Background, & its Engineering Relevance**

   - Different perspectives in Psychology. Perception and Learning: Determinants of perception, Learning theories, Behavior Modification.
   - Psychology of Adolescents: Adolescence and its characteristics
   - Motivational and Affective basis of Behavior: Basic Motives and their applications at work.
   - Components of emotions, Cognition and Emotion. Emotional Intelligence.
   - Group Dynamics and Interpersonal relationships. Development of self and personality.
   - Transactional Analysis. Culture and Mind.

2. **Module 2: Social Psychology, Ethics & Sustainability**

   - Value Spectrum for a Good Life: Role of Different Types of Values such as Individual, Societal, Material, Spiritual, Moral, and Psychological in living a good life.
   - Moral and Ethical Values: Types of Morality, Kant’s Principles of Morality, Factors for taking ethical decisions, Kohlberg’s Theory of Moral Development.
   - Analyzing individual human values such as Creativity, Freedom, Wisdom, Love and Trust.
   - Professional Ethics and Professional Ethos, Codes of Conduct, Whistle-blowing, Corporate Social Responsibility.
   - Behavioral Psychology for Sustainability: nudging eco-friendly habits, choice architecture.

3. **Module 3: Engineering Applications of Psychology (Human Factors)**

   - Human information processing model
   - Human–Machine Interaction: feedback, displays, affordances
   - User-Centered Design: principles of usability and accessibility
   - Workload & Human Error: causes, mitigation, and automation limits
   - Cognitive ergonomics and interface psychology

4. **Module 4: Motivation and Leadership in Workplace**

   - Motivation Theories: Maslow, Herzberg, McClelland in workplace context
   - Leadership Styles: autocratic, democratic, laissez-faire, transformational
   - Group Dynamics & Teamwork: team roles, stages of team development
   - Workplace Stress & Burnout: symptoms, causes, stress management techniques & Time Management & Goal Setting
   - Global Liberalisation, and its impact on Indian Economy.
   - Globalisation: Meaning, General Agreement on Trade and tariffs (GATT), World Trade Organisation (WTO).

5. **Module 5: Emotional Intelligence & Communication Skills**

   - Emotional Intelligence (EI): Daniel Goleman model, applications in leadership & teamwork
   - Interpersonal Skills: empathy, assertiveness, active listening
   - Communication Barriers: semantic, psychological, cultural
   - Non-verbal Communication: gestures, posture, tone
   - Cross-cultural Sensitivity in Global Engineering Teams

Text Books:

- Naresh Kumar, *Psychology for Engineers*, University Science Press (An imprint of Laxmi Publications)
- Robert A. Baron & Girishwar Misra, *Psychology*, Pearson India

Reference Books:

- Sanders & McCormick – *Human Factors in Engineering and Design*
- Debdulal Dutta Roy – *Applied Psychology for Engineers and Technologists*, Pearson India
- Wickens, Hollands, Banbury, & Parasuraman – *Engineering Psychology and Human Performance*, Pearson

Course Name: Digital Transformation in Business

Course Code: HU(CB)701B

Contact (Periods/Week): 3 periods

Total Contact Hours: 34 hrs

Credits: 3

Prerequisites: None

Course Objective(s):

The objective of the course is to make the students able to:

- Understand digital transformation and its business impact.
- Analyze digital technologies and business models.
- Study platform economy and data-driven strategies.
- Explore leadership, agility, and change management in digital firms.
- Evaluate challenges and strategies in digital ecosystems.

Course Outcomes (COs):

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

CO1: Define digital transformation and its key enablers.

CO2: Apply platform thinking and digital capabilities in business design.

CO3: Analyze digital strategy and customer engagement models.

CO4: Assess leadership and innovation in digitally enabled organizations.

CO5: Evaluate challenges in technology adoption and data ethics.

CO-PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	0	0	0	0	0	0	0	0	2	0	2	0
CO2	2	2	3	0	0	0	0	0	0	1	0	2	0
CO3	3	0	3	3	0	0	0	0	0	2	0	3	0
CO4	2	2	2	0	0	0	0	0	0	2	0	2	0
CO5	2	3	2	0	0	0	0	0	0	1	0	3	0

Course Contents

1. **Module 1: Introduction to Digital Business** – Transformation framework, key trends, enablers, drivers. (6L)
2. **Module 2: Digital Strategy** – Business model innovation, platforms, value chains. (8L)
3. **Module 3: Digital Technologies** – Cloud, AI, IoT, Blockchain in Fintech & Mobitech. (6L)
4. **Module 4: Leadership and Culture** – Agility, innovation, digital mindset, change management. (6L)
5. **Module 5: Risk and Ethics** – Privacy, cybersecurity, governance, tech adoption challenges. (8L)

Text Books:

- Thomas Siebel, *Digital Transformation: Survive and Thrive*, RosettaBooks, Latest Edition
- George Westerman et al., *Leading Digital*, Harvard Business Review Press, Latest Edition

Reference Books:

- Michael Wade, *Digital Strategy*, IMD
- Osterwalder & Pigneur, *Business Model Generation*
- Bain & Company, *The Lean Digital Transformation*

Course Name: Green Computing

Course Code: EE(CB)701C

Contact(Periods/Weeks): 3 Periods

Total Contact Hours: 36

Credits: 3

Prerequisites:

A fundamental understanding of IT infrastructure—covering areas like data centers, virtualization, networks, and computing systems—along with awareness of environmental sustainability concepts such as carbon footprint, green policies, and performance metrics, provides essential background for grasping the topics in this course, from Green IT fundamentals to compliance frameworks and applied case studies.

Course Outcomes (COs):

After successful completion of the course, students will be able to

CO1: Acquire knowledge to adopt green computing practices to minimize negative impacts on the environment.

CO2: Enhance the skill in energy saving practices in their use of hardware.

CO3: Evaluate technology tools that can reduce paper waste and carbon footprint by the stakeholders.

CO4: Understand the ways to minimize equipment disposal requirements.

CO5: Apply green IT strategies, compliance standards, and case study approaches to practical domains such as homes, hospitals, packaging, and telecom.

CO–PO Mapping

CO–PO–PSO Mapping

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

Course Content:

Module 1: Fundamentals (9L)
Green IT Fundamentals: Business, IT, and the Environment – Green computing: carbon footprint, scope on power – Green IT Strategies: Drivers, Dimensions, and Goals – Environmentally Responsible Business: Policies, Practices, and Metrics.

Module 2: Green Assets and Modeling (7L)

Green Assets: Buildings, Data Centres, Networks, and Devices – Green Business Process Management: Modeling, Optimization, and Collaboration – Green Enterprise Architecture – Environmental Intelligence – Green Supply Chains – Green Information Systems: Design and Development Models.

Module 3: Grid Framework (7L)

Virtualization of IT systems – Role of electric utilities – Telecommuting, teleconferencing and teleporting – Materials recycling – Best ways for Green PC – Green Data Centre – Green Grid Framework.

Module 4: Green Compliance (7L)

Socio-cultural aspects of Green IT – Green Enterprise Transformation Roadmap – Green Compliance: Protocols, Standards, and Audits – Emergent Carbon Issues: Technologies and Future.

Module 5: Case Studies (6L)

The Environmentally Responsible Business Strategies (ERBS) – Case Study Scenarios for Trial Runs – Case Studies – Applying Green IT Strategies and Applications to a Home, Hospital, Packaging Industry and Telecom Sector.

Text Books:

1. Bhuvan Unhelkar, *Green IT Strategies and Applications-Using Environmental Intelligence*, CRC Press, June 2014.
2. Woody Leonhard, Katherine Murray, *Green Home Computing for Dummies*, August 2012.

Reference Books:

1. Alin Gales, Michael Schaefer, Mike Ebbers, *Green Data Centre: Steps for the Journey*, Shroff/IBM Rebook, 2011.
2. John Lamb, *The Greening of IT*, Pearson Education, 2009.
3. Jason Harris, *Green Computing and Green IT – Best Practices on Regulations & Industry*, Lulu.com, 2008.
4. Carl Speshocky, *Empowering Green Initiatives with IT*, John Wiley & Sons, 2010.

Course Name: Big Data Analytics Lab

Course Code: CB791A

Contact (Periods/Week): 3 periods

Total Contact Hours: 36 hrs

Credits: 1.5

Prerequisites: Knowledge of Java programming, basic data structures, and database systems.

Course Objective(s):

The objective of the course is to make the students able to:

- Install, configure, and work with the Hadoop ecosystem.
- Develop MapReduce programs for real-world data processing tasks.
- Process structured and unstructured data using Pig and Hive.
- Analyze big data using Spark and build scalable applications.
- Interpret and visualize insights from various real-life big data sets.

Course Outcomes (COs):

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

CO1: Install and set up Hadoop ecosystem and run basic Hadoop commands and services.

CO2: Design and implement MapReduce programs for real-world problems involving large-scale data processing.

CO3: Analyze and manipulate structured data using Pig Latin and HiveQL queries.

CO4: Apply Spark framework for in-memory computation to solve data-intensive problems efficiently.

CO5: Evaluate real-world datasets such as MovieLens, Titanic, Uber, and analyze patterns using the Hadoop ecosystem.

CO-PO Mapping

	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PSO1	PSO2
CO1	3	2	2	2	3	0	0	0	2	2	0	3	2
CO2	3	3	3	3	3	0	0	0	2	2	0	3	3
CO3	3	2	2	3	3	0	0	0	2	2	0	3	2
CO4	3	3	3	3	3	0	0	0	3	3	0	3	3
CO5	3	3	3	3	3	0	0	0	3	3	0	3	3

Course Contents

List of Experiments:

1. Install Apache Hadoop
2. Word Frequency Count using MapReduce
3. Find Maximum Temperature Per Year using MapReduce
4. Student Grade Calculation using MapReduce
5. Matrix Multiplication using MapReduce
6. Max Electrical Consumption per Year using Monthly Data
7. Weather Dataset Analysis – Identify Cool/Shiny Day
8. Product Sales by Country using Sales Dataset
9. MovieLens Dataset: Tags associated with Movies using MapReduce
10. User Track Log Analysis (unique listeners, shares, skips, etc.) using MapReduce
11. Book Publication Frequency Per Year and Maximum Publication Year
12. Titanic Dataset Analysis – Average Age of Deaths, Survival Count by Class
13. Uber Dataset Analysis – Identify High-Trip Days by Base
14. Calculate Max Recorded Temperature per Year using Pig Latin
15. HiveQL Queries for Sorting and Aggregation
16. Develop a Java application to find Max Temperature using Apache Spark

Text Book:

- Tom White – *Hadoop: The Definitive Guide*, 4th Edition, O'Reilly Media, 2015.

Reference Books:

- Glenn J. Myatt – *Making Sense of Data*, John Wiley & Sons, 2007
- Pete Warden – *Big Data Glossary*, O'Reilly, 2011
- Michael Berthold, David J. Hand – *Intelligent Data Analysis*, Springer, 2007
- Chris Eaton et al. – *Understanding Big Data*, McGraw-Hill, 2012
- Anand Rajaraman and Jeffrey D. Ullman – *Mining of Massive Datasets*, Cambridge University Press, 2012

Course Name: Deep Learning And Neural Networks Lab

Course Code: CB791B

Contact (Periods/Week): 3 periods

Total Contact Hours: 36 hrs

Credits: 1.5

Prerequisites: Fundamentals of Python programming, machine learning concepts, and basic linear algebra.

Course Objective(s):

The objective of the course is to make the students able to:

- Provide hands-on experience with neural network development using frameworks such as TensorFlow, Keras, and PyTorch.
- Design and implement convolutional neural networks (CNNs) for image classification tasks.
- Develop deep learning models for natural language processing (NLP) applications using RNN, LSTM, and GRU.
- Experiment with advanced architectures like Autoencoders, GANs, and pre-trained models such as VGG and LeNet.
- Analyze model performance using regularization, data augmentation, and hardware acceleration (GPU).

Course Outcomes (COs):

After successful completion of the course, students will be able to:

CO1: Set up and utilize Python-based deep learning environments with TensorFlow, Keras, and PyTorch.

CO2: Design and train feedforward and convolutional neural networks for computer vision tasks.

CO3: Apply RNNs and LSTMs for sentiment analysis and natural language processing.

CO4: Implement and analyze performance of Autoencoders, GANs, and standard models like LeNet/VGG.

CO5: Optimize deep learning models using regularization, data augmentation, pruning, and batch normalization.

CO–PO Mapping

CO-PO-PSO Mapping

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

Course Contents

List of Core Experiments:

1. Set up Spyder IDE and execute a basic Python program
2. Install and explore Keras, TensorFlow, and PyTorch libraries
3. Apply CNN for computer vision tasks using image datasets
4. Image classification on MNIST with CNN and fully connected layers
5. Use deep learning for Natural Language Processing (NLP)
6. Sentiment analysis using RNN with LSTM/GRU on the IMDB dataset
7. Apply Autoencoders to compress and reconstruct data
8. Implement Generative Adversarial Networks (GANs)

Extended Project-Based Experiments:

1. Implement a single-layer Perceptron using sklearn. Analyze AND, OR, XOR datasets.
2. Implement ANN with Backpropagation and test with various activation functions.
3. Build a Deep Feed Forward ANN with ≥ 4 hidden layers and test on suitable datasets.
4. Implement a Deep Neural Network for image classification (MNIST, CIFAR-10).
5. Implement a 2-layer CNN for multi-class image classification.
6. Implement a 4-layer CNN and compare results with different epochs (5, 10, 20) on CPU and GPU.
7. Use padding and Batch Normalization in CNNs for better performance.
8. Apply L1/L2 Regularization and Dropout to CNN and compare training/test accuracy.
9. Use Data Augmentation to expand dataset from single image.
10. Apply Data Augmentation in CNN classification of CIFAR-10 and analyze results.
11. Implement LeNet-5 architecture and report accuracy on MNIST/Fashion-MNIST.
12. Implement VGG-16/VGG-19 models for image classification.
13. Use RNN for sentiment analysis of movie reviews.
14. Implement Bidirectional LSTM for sentiment analysis.
15. Generate realistic images using GANs on MNIST or face datasets.
16. Implement Autoencoders for Image Denoising using MNIST/Fashion MNIST datasets.

Text Books:

- Ian Goodfellow, Yoshua Bengio, and Aaron Courville – *Deep Learning*, MIT Press
- Franc¸ois Chollet – *Deep Learning with Python*, Manning Publications

Reference Books:

- Aur´elien G´eron – *Hands-On Machine Learning with Scikit-Learn, Keras & TensorFlow*, O’Reilly
- Rajalingappaa Shanmugamani – *Deep Learning for Computer Vision*, Packt Publishing
- Yuxi Liu – *Python Machine Learning By Example*, Packt Publishing

Course Name: Cybersecurity And Ethical Hacking Lab

Course Code: CB791C

Contact (Periods/Week): 3 periods

Total Contact Hours: 36 hrs

Credits: 3

Prerequisites: Basic networking, Linux/Windows OS familiarity, and understanding of web technologies.

Course Objective(s):

The objective of the course is to make the students able to:

- Provide hands-on exposure to foundational and advanced cybersecurity concepts.
- Equip students with practical knowledge of system-level and network-level security mechanisms.
- Demonstrate ethical hacking tools and techniques for reconnaissance, scanning, and exploitation.
- Build and test secure web applications against OWASP vulnerabilities.
- Introduce students to real-world hacking scenarios using Kali Linux and other security tools.

Course Outcomes (COs):

After successful completion of the course, students will be able to:

CO1: Understand and apply basic cybersecurity principles and tools such as encryption, antiviruses, and firewalls.

CO2: Explore Microsoft Windows internals, Active Directory components, and administrative tools.

CO3: Implement and analyze TCP/IP stack behavior, network services, and basic forensic tools.

CO4: Perform reconnaissance, scanning, vulnerability identification, and exploitation using tools like NMAP, Hydra, and BurpSuite.

CO5: Identify and exploit OWASP Top 10 vulnerabilities including XSS, SQL injection, and misconfigurations using ethical hacking techniques.

CO–PO Mapping

CO-PO-PSO Mapping

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

Course Contents

List of Modules & Practical Components:

1. **Module 1: Introduction to Cybersecurity**
   Cyber Space, Encryption, Email Security; Antivirus Tools; Career Pathways in Cybersecurity.
2. **Module 2: Microsoft OS and Active Directory**
   Windows Internals: File System, System Config, Registry, CMD; Active Directory: Domains, Forests, Users & Groups.
3. **Module 3: Networking Essentials**
   TCP/IP Layers, Ports, Firewalls; DNS, Telnet, SSH; Introduction to Network Forensics.
4. **Module 4: Ethical Hacking Basics**
   What is Hacking, Hacker Types, Google Dorks; Phases of Hacking, Passive & Active Reconnaissance; Port Scanning with NMAP & Script Engine; Exploit Search & Password Cracking Techniques.
5. **Module 5: Web Technologies**
   Website Architecture, HTML, HTTP Protocols; HTTP Methods, Headers, Cookies.
6. **Module 6: Web Application Hacking**
   Application Walkthroughs, Browser Developer Tools; Toolkits: DIRB, WhatWeb, Hydra, BurpSuite, Command Injection; XSS Theory, HTML Encoding, Vulnerability Testing Labs.
7. **Module 7: OWASP Top 10 Web Security Risks**
   Injection Attacks: SQLi, HTML Injection; Broken Access Control; Sensitive Data Exposure; Security Misconfiguration.
8. **Module 8: Advanced Hacking Tools**
   Password Cracking using John the Ripper; Video-Based Case Studies on Real-world Exploits.

Text Books:

- William Stallings – *Effective Cybersecurity: A Guide to Using Best Practices and Standards*, Pearson
- Dafydd Stuttard & Marcus Pinto – *The Web Application Hacker’s Handbook*, Wiley

Reference Books:

- Michael T. Simpson et al. – *Hands-On Ethical Hacking and Network Defense*, Cengage Learning
- Georgia Weidman – *Penetration Testing: A Hands-On Introduction to Hacking*, No Starch Press
- Jon Erickson – *Hacking: The Art of Exploitation*, No Starch Press
- Nicholas Antill – *Cybersecurity: The Beginner’s Guide*, Packt Publishing
- EC-Council – *Certified Ethical Hacker (CEH) Official Study Guide*

Course Name: Pattern Recognition Lab

Course Code: CB791D

Contact (Periods/Weeks):: 3 Periods

Total Contact Hours: 36 hrs

Credits: 1.5

Course Objective(s):

The objective of this laboratory is to provide hands-on experience in the implementation of pattern recognition algorithms. Students will gain skills in preprocessing, feature extraction, classification, clustering, and dimensionality reduction techniques, and apply them to real-life data such as images, speech, and text.

Course Outcomes (COs):

After successful completion of the lab, students will be able to:

CO1: Implement preprocessing and feature extraction techniques for different types of input data.

CO2: Design and apply supervised classification algorithms for real datasets.

CO3: Apply clustering techniques to discover hidden structures in unlabeled datasets.

CO4: Perform dimensionality reduction and evaluate the impact on recognition accuracy.

CO5: Develop complete pattern recognition pipelines for applications in image, speech, or text domains.

CO–PO Mapping

CO–PO–PSO Mapping

	PO1	PO2	PO3	PO4	PO5	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	2	–	–	3	–	–	–	–	3	–	–
CO2	–	3	3	–	2	–	–	–	–	2	3	–
CO3	–	2	–	3	2	–	–	–	–	–	3	2
CO4	–	2	–	3	–	–	–	2	–	2	2	–
CO5	–	–	3	–	3	2	2	–	3	–	3	3

Course Contents / List of Experiments

1. Introduction to Pattern Recognition Tools and Libraries (NumPy, Pandas, Scikit-learn, TensorFlow/PyTorch).
2. Feature Extraction Techniques: Implement feature extraction for image and speech data.
3. Distance Metrics: Implement Euclidean, Manhattan, and Mahalanobis distances for pattern similarity.
4. Bayes Classifier: Implement Bayes Decision Rule for binary and multiclass classification.
5. k-Nearest Neighbors (k-NN) Classifier: Implement and test with real datasets.
6. Decision Trees & Random Forests: Classification of structured datasets.

7. Support Vector Machines (SVM): Implement linear and kernel SVMs.
8. Artificial Neural Networks (ANN): Implement a simple perceptron and MLP classifier.
9. Clustering – Part I: Implement K-Means clustering for image segmentation.
10. Clustering – Part II: Implement hierarchical clustering and DBSCAN.
11. Dimensionality Reduction – PCA: Apply PCA on datasets and analyze performance.
12. Dimensionality Reduction – LDA: Apply Linear Discriminant Analysis for classification tasks.
13. Case Study I – Image Recognition: End-to-end system for digit/face recognition.
14. Case Study II – Speech/Text Recognition: Implement a simple speech command/text classification system.
15. Mini-Project: Develop a complete pattern recognition system integrating feature extraction, classification, and evaluation.

Assessment Method

- Continuous Evaluation (Internal): Lab performance, quizzes, and viva (30%)
- Lab Records/Assignments: Implementation reports and submissions (20%)
- Mini-Project / End-Sem Evaluation: Comprehensive implementation and presentation (50%)

Textbooks / References:

1. Christopher M. Bishop, *Pattern Recognition and Machine Learning*, Springer Verlag, 2006.
2. Trevor Hastie, Robert Tibshirani, Jerome Friedman, *The Elements of Statistical Learning*, Springer, 2009.
3. Ulisses Braga-Neto, *Fundamentals of Pattern Recognition and Machine Learning*, Springer, 2020.
4. A.K. Jain and R.C. Dubes, *Algorithms for Clustering Data*, Prentice Hall, 1988.
5. Jiawei Han, Micheline Kamber, *Data Mining: Concepts and Techniques*, Morgan Kaufmann, 2011.
6. Ian Goodfellow, Yoshua Bengio, Aaron Courville, *Deep Learning*, MIT Press, 2016.

Course Name: Natural Language Processing Lab

Course Code: CB792A

Contact (Periods/Week): 3 periods

Total Contact Hours: 36 hrs

Credits: 3

Prerequisites: Basic programming in Python, and foundational knowledge of machine learning and probability.

Course Objective(s):

The objective of the course is to make the students able to:

- Introduce basic NLP concepts using Python and NLTK.
- Practice essential NLP tasks such as tokenization, stemming, lemmatization, and tagging.
- Perform parsing, chunking, and named entity recognition using standard NLP tools.
- Implement real-world applications such as spam detection, fake news detection, and sentiment analysis.
- Use industry-level NLP tools such as the Stanford NER and evaluate corpus-based frequency analysis.

Course Outcomes (COs):

After successful completion of the course, students will be able to:

CO1: Apply fundamental NLP techniques like tokenization, stemming, lemmatization, and POS tagging using NLTK.

CO2: Implement named entity recognition, chunking, parsing and TF-IDF based analysis.

CO3: Analyze language models and perform n-gram-based probability estimation.

CO4: Perform real-world NLP applications such as spam filtering, fake news detection, and sentiment analysis.

CO5: Evaluate corpus-based patterns using TF-IDF, document statistics, and Stanford NER.

CO–PO Mapping

CO-PO-PSO Mapping

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

Course Contents

List of Experiments:

1. **Week 1:** Tokenization by word and sentence using NLTK; Stopword elimination; Stemming.
2. **Week 2:** POS Tagging; Lemmatization.
3. **Week 3:** Chunking; Named Entity Recognition (NER).
4. **Week 4:** TF-IDF Calculation; CYK / Chart Parsing.
5. **Week 5:** Generating Unigrams, Bigrams, Trigrams; Sentence Probability Estimation.
6. **Week 6:** Entity extraction using Stanford NER programmatically.
7. **Week 7:** Stopword frequency count, TF-IDF, and importance analysis using a corpus.
8. **Week 8:** Sentiment Analysis using NLP.
9. **Week 9:** Spam Filter development using NLP.
10. **Week 10:** Fake News Detection using NLP.

Text Books:

- Steven Bird, Ewan Klein, and Edward Loper – *Natural Language Processing with Python*, O'Reilly
- Laura Igual and Santi Segu´ı – *Introduction to Data Science*, Springer

Reference Books:

- Daniel Jurafsky and James H. Martin – *Speech and Language Processing*, Pearson
- Jacob Eisenstein – *Introduction to Natural Language Processing*, MIT Press
- Yoav Goldberg – *Neural Network Methods in Natural Language Processing*, Morgan & Claypool
- Ekaterina Kochmar – *Practical Natural Language Processing*, O'Reilly

Course Name: Introduction To Blockchain Lab

Course Code: CB792B

Contact (Periods/Week): 3 periods

Total Contact Hours: 36 hrs

Credits: 3

Prerequisites: Basic knowledge of data structures, cryptography, and programming (preferably Python or Solidity).

Course Objective(s):

The objective of the course is to make the students able to:

- Understand the working principles of blockchain data structures and cryptographic components.
- Implement blockchain mechanisms such as block creation, Merkle trees, and mining.
- Explore smart contracts and token standards like ERC-20.
- Demonstrate peer-to-peer (P2P) networking in a blockchain environment.
- Build and simulate basic cryptocurrency wallets and transaction flows.

Course Outcomes (COs):

After successful completion of the course, students will be able to:

CO1: Implement core blockchain data structures including Merkle Trees and Blocks.

CO2: Develop and simulate a basic blockchain and integrate Merkle tree logic within it.

CO3: Create and deploy ERC-20 tokens and smart contracts.

CO4: Simulate mining and peer-to-peer architecture within a blockchain network.

CO5: Design a basic cryptocurrency wallet with essential transaction functionality.

CO–PO Mapping

CO-PO-PSO Mapping

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

Course Contents

List of Experiments:

1. Creating Merkle Tree – Implementing hash-based structure to secure transactions.
2. Creation of Block – Creating data blocks with hash, nonce, and transaction record.
3. Blockchain Implementation – Linking blocks using cryptographic hashes.
4. Creating ERC-20 Token – Creating a standard Ethereum-based token.
5. Blockchain using Merkle Trees – Integrating Merkle trees into blockchain.
6. Mining in Blockchain – Simulating proof-of-work and block rewards.
7. Peer-to-Peer Blockchain Network – Setting up P2P communication between nodes.
8. Creating Cryptocurrency Wallet – Building a basic wallet with transaction features.

Text Books:

- Andreas M. Antonopoulos – *Mastering Bitcoin: Unlocking Digital Cryptocurrencies*, O'Reilly Media
- Imran Bashir – *Mastering Blockchain*, Packt Publishing

Reference Books:

- Arvind Narayanan et al. – *Bitcoin and Cryptocurrency Technologies*, Princeton University Press
- Joseph Bonneau – *SoK: Research Perspectives and Challenges for Bitcoin and Cryptocurrencies*
- Roger Wattenhofer – *The Science of the Blockchain*, Inverted Forest Publishing
- Narayanan, Bonneau et al. – *Bitcoin and Blockchain Technology Explained*, Princeton Lectures
- Gavin Wood – *Ethereum: A Secure Decentralised Generalised Transaction Ledger (Yellow Paper)*

Course Name: Introduction to Game Development Lab

Course Code: CB792D

Contact (Periods/Week): 3 periods

Total Contact Hours: 36

Credits: 1.5

Course Objectives

- To provide hands-on experience with game engines and development tools.
- To enable students to implement basic game programming concepts in 2D and 3D environments.
- To develop skills in integrating graphics, physics, audio, and user interaction in games.
- To encourage teamwork and creativity through mini-projects in game design.
- To expose students to testing, debugging, and deployment practices in game development.

Course Outcomes (COs)

- **CO1:** Install and configure game development tools and environments.
- **CO2:** Develop interactive 2D and 3D games using basic programming constructs.
- **CO3:** Apply physics, animation, and audio in enhancing gameplay experience.
- **CO4:** Design user interfaces and integrate assets effectively in game projects.
- **CO5:** Work collaboratively to design, implement, and present a mini-game project.

CO–PO Mapping

CO-PO-PSO Mapping

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

List of Suggested Experiments

1. Installation and setup of a game engine (Unity / Unreal / Godot).
2. Creating a simple “Hello World” game project.
3. Designing 2D game scenes: sprites, backgrounds, and animations.
4. Implementing player controls and event-driven interactions.

5. Physics in games: collision detection, gravity, and rigid body dynamics.
6. Adding audio effects and background music to games.
7. Implementing UI elements: menus, buttons, and scoreboards.
8. Creating a simple 3D environment and controlling a camera.
9. Scripting gameplay mechanics such as scoring and health.
10. Integrating assets and prefabs for rapid prototyping.
11. Developing a simple multiplayer/local co-op feature.
12. Mini-project: design and implement a small interactive game.

Textbooks

1. Joseph Hocking, *Unity in Action: Multiplatform Game Development*, Manning.
2. David Nixon, *Learning C# by Developing Games with Unity*, Packt.

Reference Books

1. Robert Nystrom, *Game Programming Patterns*, Genever Benning.
2. Ernest Adams, *Fundamentals of Game Design*, Pearson.

4th Year 8th Semester									
Sl. No.	**Broad Category**	**Category**	**Paper Code**	**Subject**	**L**	**T**	**P**	**Total**	**Credit Points**
A. Theory									
B. Practical									
1	PRJ	PRJ	CB881	Internship or Entrepreneurship	0	0	0	0	3
2	ENGG	Major	CB882	Grand Viva	0	0	0	0	2
Total of Theory, Practical								**0**	**5**