SCHEME OF INSTRUCTIONS AND SYLLABUS

B.Tech. in Computer Science and Engineering
(Artificial Intelligence and Date Science)
Regulation: SU24

w.e.f. 2024-25

Departement of Computer Science and Engineering

Velagapudi Ramakrishna
Siddhartha School of Engineering
SIDDHARTHA ACADEMY OF HIGHER EDUCATION
(An Institution Deemed to be University)
Under section 3 of UGC Act 1956
Sponsered by Siddhartha Academy of General & Technical Education

INSTITUTION VISION

To be a Centre of Excellence in Education, Innovation and Research with Global presence in Arts, Science, Technology, Medicine, Management, Legal and Social Studies in enriching the frontier areas of National and International Importance.

INSTITUTION MISSION

- To create a transformative educational experience for students focused on problem solving skills; communication abilities, and interpersonal relations and leadership.
- To cultivate a vibrant university community for attracting and retaining diverse, world-class talent creating a collaborative environment open to the free exchange of ideas where research, creativity, innovation and entrepreneurship can flourish and ensuring individuals to achieve their full potential
- To impact society in a pragmatic manner— regionally, nationally, and globally — by engaging with industry, outstanding national and international universities and research organizations
- To be a global University that nurtures excellence in education and innovation for creating a knowledgeable society

DEPARTMENT VISION

To evolve as a centre of academic excellence and advanced research in Computer Science and Engineering discipline

DEPARTMENT MISSION

To inculcate students with profound understanding of fundamentals related to discipline, attitudes, skills, and their application in solving real world problems, with an inclination towards societal issues and research

PROGRAM EDUCATIONAL OBJECTIVES (UG)

We have program educational objectives for our Computer Science and Engineering Program. Program educational objectives are broad statements that describe the career and professional accomplishments that the program is preparing graduates to achieve.

Our program objectives are:

- The graduates of the Program will have solid foundation in the principles and practices of computer science, including mathematics, science and basic engineering.
- The graduates of the Program will have skills to function as members of multi-disciplinary teams and to communicate effectively using modern tools.
- The graduates of the Program will be prepared for their careers in the software industry or pursue higher studies and continue to develop their professional knowledge.
- The graduates of the program will practice the profession with ethics, integrity, leadership and social responsibility.

PROGRAM OUTCOMES

On successful completion of the B.Tech CSE Programme the student will be able to :

- **PO1:Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.
- **PO2: Problem analysis:** Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.
- PO3: Design/development of solutions: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.
- **PO4: Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
- **PO5: Modern tool usage:** Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
- **PO6: The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
- **PO7: Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.
- **PO8: Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
- **PO9: Individual and team work:** Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.
- **PO10: Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.
- **PO11: Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
- **PO12: Lifelong learning:** Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

PROGRAM SPECIFIC OUTCOMES

- **PSO1:** Develop AI based software applications/solutions as per the needs of Industry and Society.
- **PSO2:** Adopt new and fast emerging technologies in Artificial Intelligence and Data Science.

SIDDHARTHA ACADEMY OF HIGHER EDUCATION
(Deemed to be University)
Velagapudi Ramakrishna
Siddhartha School of Engineering
SU24 Regulations
Department of Computer Science and Engineering
B.Tech. Computer Science and Engineering
(Artificial Intelligence and Data Science)
Curriculum for Semesters I to VIII

The B.Tech – CSE (AI & DS) program focuses on providing a strong foundation in artificial intelligence and data science, two rapidly evolving and impactful fields. The curriculum integrates core computer science concepts with specialized topics like m**achine learning, data mining, big data analytics, deep learning, predictive analytics, and natural language processing**, ensuring students develop a comprehensive skill set in both artificial intelligence and data analytics. The department is supported by a team of expert faculty members actively engaged in research, and it provides state-of-the-art infrastructure, including advanced labs, software tools, and cloud platforms for hands-on learning. With a strong emphasis on research, industry collaboration, and real-world problem-solving, students have ample opportunities for internships, projects, and competitions. Graduates are prepared for high-demand careers as **data scientists, AI engineers, data analysts,** and other roles in industries such as technology, healthcare, finance, and retail.

SEMESTER-I **CONTACT HOURS:28**

S.No	Course Code	Course Category	Course Name	L	T	P	Credits
1	24MA101	BS	Linear Algebra, Series, and Calculus	3	0	2	4
2	24PH101	BS	Applied Physics	3	0	0	3
3	24EN101	HS	Communicative English	3	0	3	4.5
4	24CS101	ES	**Programming using C**	3	0	0	3
5	24UC101	ES	Design Thinking	0	0	2	1
6	24PH181	BS	Physics Lab	0	0	2	1
6	24CS182	ES	**Programming using C Lab**	0	0	3	1.5
7	24UC182	ES	AI Tools and Applications	0	0	2	1
8	24UC101	MC	Essence of Indian Knowledge Tradition	2	0	0	—
Total				**14**	**0**	**14**	**19**

SEMESTER-II **CONTACT HOURS:31**

S.No	Course Code	Course Category	Course Name	L	T	P	Credits
1	24MA102	BS	Differential Eqs. & Numerical methods	3	0	2	4
2	24CY101	BS	Applied Chemistry	3	0	0	3
3	24IT101	ES	**Problem Solving with Python**	2	0	2	3
4	24IT102	ES	Data Structures	2	0	2	3
5	24BY101	BS	Biology for Engineers	3	0	0	3
6	24ME181	ES	Engineering Graphics	1	0	3	2.5
7	24CY181	BS	Chemistry Lab	0	0	2	1
8	24ME182	ES	Workshop Practice	0	0	3	1.5
9	24UC183	MC	Sports & Yoga or NSS or NCC	0	0	3	—
Total				**14**	**0**	**17**	**21**

SEMESTER-III **CONTACT HOURS:27**

S.No	Course Code	Course Category	Course Name	L	T	P	Credits
1	24MA203	BS	Probability and Statistics	4	0	0	4
2	24CS201	ES	Digital Logic and Design	3	0	0	3
3	24CS202	PC	Object Oriented Programming Through Java	3	0	0	3
4	24CS203	PC	Software Engineering	3	0	0	3
5	24CS204	PC	Operating Systems	3	0	2	4
6	24CS281	PC	Object Oriented Programming Through Java Lab	0	0	3	1.5
7	24CS282	ES	Digital Logic and Design Lab	0	0	3	1.5
8	24UC201	MC	Universal Human Values - II	2	1	0	3
Total				**18**	**1**	**8**	**23**

SEMESTER-IV **CONTACT HOURS:29**

S.No	Course Code	Course Category	Course Name	L	T	P	Credits
1	24MA204	BS	Discrete Mathematics	3	0	0	3
2	24CS205	PC	Database Management Systems	3	0	0	3
3	24CS206	PC	Design and Analysis of Algorithms	3	0	0	3
4	24CS207	PC	Computer Organization & Architecture	3	0	2	4
5	24CS209	PC	Artificial Intelligence	3	0	2	4
6	24CS283	PL	Database Management Systems Lab	0	0	3	1.5
7	24CS284	PL	Design and Analysis of Algorithms Lab	0	0	3	1.5
8	24EN281	HS	English for Professionals	0	0	2	1
9	24UC202	MC	Professional Ethics	2	0	0	0
Total				**17**	**0**	**12**	**21**

- **Engineering Project in Community Services (EPICS)**: The Engineering Project for community services will be carried out during summer vacation for a period of six weeks after the IV Semester and the report shall be submitted in the V Semester. Students will go to the society (Villages/ Hospitals/Towns, etc..) to identify the problem, survey the literature and discuss with the community for a feasible solution. The students are encouraged to solve real-life problems.
- **Mini Project:** The Mini Project is carried out during the VII semester. Students have to carry out feasibility studies, and literature surveys, and prepare a detailed project report.
- **Major Project:** The Major Project is carried out in the VIII semester and the student can carry out his/her project work in an industry/R&D organization/in the college with well-defined objectives. At the end of the semester, the student shall submit a detailed project report. It involves the preparation and presentation of a report and students are encouraged to publish their work in any research journal/conference. The project report shall be evaluated by a committee appointed by HoD.

SEMESTER-V **CONTACT HOURS:32**

S.No	Course Code	Course Category	Course Name	L	T	P	Credits
1	24AIDS301	PC	Machine Learning	3	0	0	3
2	24AIDS302	PC	Computer Networks	3	0	0	3
3	24AIDS303	PC	Theory of Computation	3	0	0	3
4	24AIDS360A	IE	Inter disciplinary Elective I	3	0	0	3
5	24AIDS310A	PE	ProgramElective-1 A. Big Data Analytics B. Information Retrieval C. R Programming D. Computer Graphics E. UI & UX Design F. Full Stack Development-I	3	0	0	3
6	24AIDS303	PL	Machine Learning Lab	0	0	3	1.5
7	24AIDS384	PL	Computer Networks Lab	0	0	3	1.5
8	24AIDS385	PR	EPICS	0	0	4	2
9	24AIDS381	HS	Advanced communication skills	0	0	2	1
10	24AIDS382	PC	Industry Standard Coding Practice - I	1	0	2	2
11	24UC301	MC	Constitution of India	2	0	0	0
Total				**18**	**0**	**14**	**23**

SEMESTER-VI **CONTACT HOURS:27**

S.No	Course Code	Course Category	Course Name	L	T	P	Credits
1	24AIDS304	PC	Data Science	3	1	0	4
2	24AIDS305	PC	Data Visualization	3	0	0	3
3	24AIDS370A	IE	Inter disciplinary Elective–II	3	0	0	3
4	24AIDS320A 24AIDS320B 24AIDS320C 24AIDS320D 24AIDS320E 24AIDS320F	PE	Program Elective-2 A. Time Series Analysis & Forecasting B. Remote Sensing and GIS C. Exploratory Data Analysis D. Digital Image Processing E. Cryptography & Network Security F. Full Stack Development-II	3	0	0	3
5	24AIDS330A 24AIDS330B 24AIDS330C 24AIDS330D 24AIDS330E 24AIDS330F	PE	Program Elective-3 A. Graph Analytics B. Predictive Analytics C. Computational Neuroscience D. Augmented and Virtual Reality E. Cyber Forensics F. Mobile Application Development	3	0	0	3
6	24AIDS386	PL	Data Science Lab	0	0	3	1.5
7	24AIDS387	PL	Data Visualization Lab	0	0	3	1.5
8	24AIDS383	PC	Industry Standard Coding Practice-II	1	0	2	2
Total				**16**	**1**	**8**	**21**

SEMESTER-VII **CONTACT HOURS:25**

S.No	Course Code	Course Category	Course Name	L	T	P	Credits
1	24AIDS401	PC	Compiler Design	2	0	2	3
2	24OE410A	OE	Open Elective– I	3	0	0	3
3	24AIDS440A 24AIDS440B 24AIDS440C 24AIDS440D 24AIDS440E 24AIDS440F	PE	Program Elective-4 A. Social Network Analytics B. Web Analytics C. Recommender Systems D. Text and Speech Analysis E. Cyber Security F. Full Stack Development-III	3	0	0	3
4	24AIDS410A	HE	Humanities Elective – I	3	0	0	3
5	24AIDS481	PR	Summer Internship	0	0	4	2
6	24AIDS482	PC	Advanced Skill Course	1	0	2	2
7	24AIDS483	PR	Mini Project	0	0	4	2
8	24UC401	MC	Professional Ethics	2	0	0	—
Total				**14**	**0**	**12**	**18**

SEMESTER-VIII **CONTACT HOURS:22**

S.No	Course Code	Course Category	Course Name	L	T	P	Credits
1	24HE420A	HE	Humanities Elective -II	3	0	0	3
2	24OE420A	OE	Open Elective– II	3	0	0	3
3	24AIDS484	PR	Major Project	0	0	16	16
Total				**6**	**0**	**16**	**14**

Humanities Elective Courses (HE Basket)
Project Management
Engineering Economics & Mgmt.
Innovation, IPR & Enterpreneurship
Operations Research
Industrial Psychology
Finance and Accounting
Organizational Behavior
+
Any management course offered by SAHE/Online (Swayam/NPTEL) and approved by department

Open Elective Courses (OE Basket)
Foreign Language
Music
Law
+
Any course offered by SAHE/Online (Swayam/NPTEL) and approved by department

COURSE CODE: 24MA203
PROBABILITY AND STATISTICS

Course Category:	Basic Sciences (BS)	**Credits:**	4
Course Type:	Theory	**Lecture -Tutorial-Practice:**	4-0-0
Pre-requisites:	Linear Algebra, Series, and Calculus	**Continuous Evaluation:** **Semester end Evaluation:** **Total Marks:**	40 60 100

Course Description

This course provides a comprehensive understanding of probability and statistics, focusing on theoretical foundations and practical applications. Students will explore fundamental concepts such as probability distributions, statistical inference, hypothesis testing, and data analysis techniques. Key topics include discrete and continuous probability distributions (Binomial, Poisson, Normal), joint distributions, and the Central Limit Theorem. Emphasizing real-world applications, students will develop problem-solving skills to analyze data, model uncertainties, and make informed decisions based on statistical reasoning.

Course Objectives

- To develop a strong foundation in probability theory, including discrete, continuous, and joint distributions.
- To learn to construct confidence intervals, perform hypothesis testing, and make data-driven inferences.
- To gain proficiency in statistical estimation, variance analysis, and data interpretation techniques.
- To apply probability and statistical models to real-world problems in machine learning, finance, and research.
- To use computational tools to implement probability and statistical techniques for data analysis and decision-making.

Course Outcomes

At the end of the course, the student will be able to

	Course Outcomes	BTL	POI
CO1	Explain probability distributions, joint distributions, and the Central Limit Theorem.	**K2**	1.2.1, 1.2.2, 2.6.3, 4.6.1
CO2	Interpret confidence intervals and hypothesis testing for statistical inference.	**K2**	1.2.1, 1.2.2, 2.6.3, 4.6.1
CO3	Apply probability and statistical methods to real-world problems.	**K3**	1.2.2, 2.6.3, 4.6.1, 4.6.3
CO4	Analyze datasets to estimate parameters and test hypotheses.	**K3**	2.5.2, 2.6.3,3.5.1
CO5	Evaluate probabilistic models using computational tools for decision-making.	**K4**	1.2.1, 2.6.3, 4.6.1, 4.6.3

Contribution of Course Outcomes towards achievement of Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2
CO1	3	3		2										
CO2	3	3		2									1	
CO3	3	3		2									1	
CO4	3	3		2									1	
CO5	3	3		2									1	

Course Content

Unit – I :Probability Distributions and Random Variables

Probability Distributions: Random Variables, Binomial distribution, Hypergeometric Distribution, Poisson approximation to the Binomial distribution, Poisson process.

Unit – II: Probability Densities and Joint Distributions

Probability Densities: Continuous random variables, Normal distribution, Normal approximation to the Binomial distribution.
Joint distribution: Joint Distributions-Discrete and Continuous.

Unit – III: Statistical Inference

Inferences Concerning a Mean: Point Estimation- Interval Estimation, Tests of Hypotheses, Null Hypotheses and Tests of Hypotheses, Hypotheses concerning one mean, Relation between tests and confidence intervals, Comparisons-Two independent large samples, Comparisons-Two independent small samples.

Unit – IV :Variances and Proportions

Inferences Concerning Variances: Estimation of variances, Hypotheses concerning one variance, Hypotheses concerning two variances.
Inferences Concerning Proportions: Estimation of Proportions, Hypotheses concerning one Proportion, Hypotheses concerning several Proportions, The analysis of r x c tables.

Unit – V: Non-Parametric Statistical Tests

Non parametric Tests: Introduction, The Sign Test, Rank-Sum tests, Correlation based on ranks, Tests of Randomness.

Text Books

1. R. A. Johnson, Probability and Statistics for Engineers, 8th Edition, Prentice Hall India Learning Private Limited, 2011.

Reference Books

1. R. E. Walpole, R. H. Myers, S. L. Myers, and K. Ye, Probability and Statistics for Engineers and Scientists, 5th Edition, Macmillan, New York, 1993.
2. P. C. Biswal, Probability and Statistics, Prentice Hall India Learning Private Limited, 2007.
3. T. K. V. Iyengar, B. K. Gandhi, S. Ranganadham, and M. V. S. S. N. Prasad, Probability and Statistics, S. Chand Publishing, 2008.

Web Resources

1. **NPTE**L, ""Probability and Statistics," Prof. Somesh Kumar, Department of Mathematics, IIT Kharagpur, [Online]. Available: `https://nptel.ac.in/courses/111105090`. [Accessed: Mar. 7, 2025].
2. **NPTEL**, "Probability and Statistics," Prof. Niladri Chatterjee, Department of Mathematics, IIT Delhi, [Online]. Available: `https://nptel.ac.in/courses/111102112`. [Accessed: Mar. 7, 2025].
3. **NPTEL**, "Probability and Statistics," Prof. Soumen Maity, Department of Mathematics, IISER Pune, [Online]. Available: `https://nptel.ac.in/courses/111105042`. [Accessed: Mar. 7, 2025].

COURSE CODE: 24CS201
DIGITAL LOGIC DESIGN

Course Category:	Engineering Sciences (ES)	**Credits:**	3
Course Type:	Theory	**Lecture -Tutorial-Practice:**	3-0-0
Pre-requisites:	—	**Continuous Evaluation:** **Semester end Evaluation:** **Total Marks:**	40 60 100

Course Description

Digital Logic Design provides an in-depth introduction to the principles and techniques used in the design and analysis of digital systems. Students will explore the fundamentals of digital logic, the building blocks of modern computing systems, and their applications in hardware design. The course emphasizes both combinational and sequential logic circuits, digital number systems, Boolean algebra, analysis, design, and evaluation of digital circuits, of medium complexity, that are based on SSIs, MSIs, and programmable logic devices. The main objectives are to provide knowledge on methods for simplifying Boolean functions and to develop skills for design of various combinational & sequential logic circuits.

Course Objectives

- To introduce key number systems, Boolean algebra, and logic gate concepts required for digital system design.
- To simplify Boolean functions using various methods and apply them in designing efficient digital systems.
- To design and implementation of combinational and sequential logic circuits.
- To develop an understanding of the structure and operation of programmable logic devices and their applications in system design.
- To design, analyze, and implement memory systems and counters.

Course Outcomes

At the end of the course, the student will be able to

	Course Outcomes	BTL	POI
CO1	Understand Number systems and Boolean algebra.	**K2**	1.2.1, 1.7.1
CO2	Apply simplification techniques to minimize Boolean functions.	**K3**	1.2.1, 1.7.1, 2.7.2
CO3	Apply combinational logic concepts to design and implement MSI and LSI circuits	**K3**	1.7.1, 2.7.1, 2.7.2, 3.7.1 ,3.8.1
CO4	Apply sequential logic principles to design clocked sequential circuits using flip-flops.	**K3**	1.7.1, 2.7.1, 2.7.2, 3.7.1 ,3.8.1
CO5	Apply sequential logic techniques to design Counters, Registers and Memory elements.	**K3**	1.7.1, 2.7.1, 2.7.2, 3.7.1 ,3.8.1

Contribution of Course Outcomes towards achievement of Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2
CO1	2													
CO2	2	2												
CO3	2	2	2											1
CO4	2	2	2											1
CO5	1	2	2											

Course Content

Unit – I : Number Systems, Boolean Algebra, and Logic Gates

Binary Systems: Digital computers and digital systems, Number systems, conversions Complements: r's complement, (r-1)'s complement, Binary Codes, Representation of integers and Floating-point numbers, Introduction to integer arithmetic operations.
Boolean Algebra and Logic Gates: Basic Definitions, Axiomatic definition of Boolean Algebra, Basic theorems and Properties of Boolean Algebra, Boolean functions, Canonical and Standard Forms, Other logic operations, Digital Logic Gates

Unit – II : Boolean Function Simplification and Combinational Logic Design

Simplification of Boolean Functions: The Map Method, Two and three variable Maps, Four-variable Map, five variable Map, Product of Sums Simplification, Don't care conditions, The Tabulation Method, Determination of Prime Implicants, Selection of Prime-Implicants.
Combinational Logic: Introduction, Design Procedure, Adders, Subtractors, Code Conversion, Analysis Procedure. Exclusive-or Gates, Parity Generators and Checkers.

Unit – III : Combinational Circuits and Programmable Logic Devices

Combinational Logic with MSI and LSI: Binary Parallel Adder, Decimal Adder, Magnitude Comparator, Decoders, demultiplexers, encoders, Multiplexers.
Programmable Logic: Programmable Logic Devices (PLD), Programmable read only memory (PROM), Programmable Logic Array (PLA), Programmable Array Logic (PAL), Introduction to FPGA.

Unit – IV : Sequential Logic and State Machine Design

Sequential Logic: Sequential circuits, Classification, Latches, Flip Flops, Triggering of Flip-Flops, Master slave flip-flop, Flip-Flop Excitation tables, flip-flop direct inputs.
Analysis of Clocked Sequential Circuits: State table, State diagram, state equations, State Reduction and Assignment, Design Procedure, design with unused states, Design of Counters.

Unit – V : Registers, Counters, and Memory Units

Registers, Counters: Registers, Shift Registers, Asynchronous Counters, Synchronous Counters, Ring Counter, Johnson Counter, Timing Sequences.
Memory Units: Block diagram of memory unit, Design of ROM, Classification of ROMs, Design of RAM, Classification of RAMs.

Text Books

1. M.Morris Mano, Digital Logic & Computer Design 1 e/d reprint, Pearson education, 2016.
2. M.Morris Mano, Michael D Ciletti Digital Design with an Introduction toVerilog HDL 5th e/d, Pearson education, 2013

Reference Books

1. A. Anand Kumar, Switching Theory and Logic Design, 2nd Edition, PHI,2013
2. Charles H.Roth , Fundamentals of Logic Design, 6/e, Cengage learning,2010
3. Computer Architecture and Organization Designing for Performance, William Stallings, Ninthedition, Pearson Education series, 2014.

Web Resources

1. **NPTEL**, "Digital Systems," Prof. N. Goel, Department of Electrical Engineering, IIT Ropar, [Online]. Available: `https://nptel.ac.in/courses/108/106/108106177/`. [Accessed: Feb. 25, 2025].
2. **NPTEL**, "Digital Systems," Prof. N. J. Rao, Department of Electrical Engineering, IISc Bangalore, [Online]. Available: `https://nptel.ac.in/courses/106/108/106108099/`. [Accessed: Feb. 25, 2025].

COURSE CODE: 24CS202
OBJECT ORIENTED PROGRAMMING THROUGH JAVA

Course Category:	Program Core (PC)	**Credits:**	3
Course Type:	Theory	**Lecture -Tutorial-Practice:**	3-0-0
Pre-requisites:	Programming using 'C'	**Continuous Evaluation:** **Semester end Evaluation:** **Total Marks:**	40 60 100

Course Description

The course provides a comprehensive introduction to Java programming, focusing on fundamental constructs, object-oriented principles, and advanced features. It begins with the evolution of Java, basic syntax, data types, and control structures. Students will explore classes, objects, inheritance, and string handling to build modular and reusable code. The course further covers packages, interfaces, exception handling, and file I/O operations to enhance application robustness. Advanced topics include multithreading for concurrent execution and lambda expressions for functional-style programming. The course concludes with the Java Collections Framework and Stream API, enabling efficient manipulation of object groups and data processing. Emphasis is placed on applying Java features to solve real-world programming problems effectively.

Course Objectives

- To understand the fundamentals of Java programming, including its evolution, key features, object-oriented principles, control structures, and basic data handling using variables, operators, and arrays.
- To apply object-oriented concepts like classes, objects, inheritance, and string handling to develop modular Java programs.
- To apply packages and interfaces for modular programming, manage runtime errors through exception handling, and perform file input/output operations using byte and character streams in Java.
- To apply Java multithreading and lambda expression features for building efficient, concurrent, and functional programs.
- To apply the principles of Java Collections Framework and Stream API in developing programs that effectively manage and operate on groups of objects.

Course Outcomes

At the end of the course, the student will be able to

	Course Outcomes	BTL	POI
CO1	Understand Java's history, features, object-oriented concepts, control statements, and basic data types.	**K2**	1.7.1, 2.5.1, 3.5.6
CO2	Apply concepts of classes, objects, inheritance, and string handling to develop Java applications.	**K3**	1.7.1, 2.6.3, 3.5.1
CO3	Apply the concepts of packages, interfaces, exception handling, and I/O streams to develop efficient and modular Java programs.	**K3**	1.7.1, 2.5.2, 3.5.1, 5.4.1
CO4	Apply multithreading and lambda expressions in Java to develop efficient, concurrent, and functional programs.	**K3**	1.7.1, 2.7.2, 3.8.2, 5.4.2
CO5	Apply Java Collections Framework and Stream API for managing groups of objects and performing efficient operations on them.	**K3**	1.7.1, 2.7.1, 3.7.1, 5.4.1

Contribution of Course Outcomes towards achievement of Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2
CO1	2	3	1										1	
CO2	2	3	1										2	2
CO3	1	2	2		3								2	2
CO4	1	2	3		2								2	2
CO5	2	2	1		3								2	2

Course Content

Unit – I : Fundamental Java Programming Constructs

Introduction, The History and Evolution of Java: Java History and Evolution, Java Features, Java's Magic: Byte Code, How Java differs from C and C++.
An Overview of Java: Object Oriented Programming, Two Paradigms, Principles of OOP, A First Simple Program, Two Control Statements, Java keywords.
Data Types, Variables and Arrays: The Primitive Types, Integers, Floating-Point Types, Characters, Booleans, Variables, Type Conversion and Casting, Automatic Type Promotion in Expressions, Arrays, Operators.

Unit – II : Classes, Objects, Inheritance, and String Handling

Introducing Classes and Objects: Class Fundamentals, Declaring Objects, Assigning Object Reference Variables, Introducing Methods, Constructors, The this keyword, Garbage Collection, Overloading Methods, Using Objects as Parameters, Returning Objects, Understanding static, Introducing final, Introducing Nested and Inner Classes.
String Handling: The String Constructors, StringBuffer Class, StringTokenizer class.
Inheritance: Inheritance Basics, Using super, Creating a Multilevel Hierarchy, Method Overriding, Dynamic Method Dispatch, Using Abstract Classes, Using final with Inheritance.

Unit – III : Packages and Interfaces, Exception Handling, and IO Streams

Packages & Interfaces: Packages, Defining a Package, Finding Package and CLASSPATH, A Short Package Example, Packages and Member Access, Importing Packages, Interfaces, Defining an Interface, Implementing Interfaces, Nested Interfaces, Applying Interfaces, Variables in Interfaces.
Exception handling: Exception Handling Fundamentals, Exception Types, Uncaught Exceptions, Using try and catch, Multiple catch Clauses, throw, throws, finally, Creating Your Own Exception Subclasses
I/O streams: The Byte Streams - InputStream, OutputStream, FileInputStream, FileOutputStream, The Character Streams - Reader, Writer, FileReader, and FileWriter.

Unit – IV : Multithreading and Lambda Expressions

Multithreaded Programming: The Java Thread Model, The Main Thread, Creating a Thread, Creating Multiple Threads, Using isAlive() and join(), Thread Priorities, Synchronization.
Lambda Expressions: Introducing Lambda Expressions, Block Lambda Expressions, Generic Functional Interfaces, Passing Lambda Expressions as Arguments, Lambda Expressions and Exceptions, Lambda Expressions and Variable Capture, Method References.

Unit – V : Collections Framework and Stream API

Collections Framework: Collections Overview, List Interface, Set Interface, Map Interface, ArrayList Class, LinkedList Class, HashSet Class, HashMap Class, TreeMap Class.
The Stream API: Stream Basics, Reduction Operations, Using Parallel Streams, Mapping, Collecting, Iterators and Streams.

Text Books

1. Herbert Schildt, Danny Coward, “Java: The Complete Reference”, Thirteenth Edition, McGrawHill, 2023.

Reference Books

1. Herbert Schildt, Dale Skrien, “Java Fundamentals A Comprehension Introduction”, Special Indian Edition, McGraw-Hill Education India Pvt. Ltd, 2017.
2. E Balaguruswamy, ”Programming with Java”, Seventh Edition, Mc Graw Hill 2023.
3. Paul J. Dietel and Dr.Harvey M. Deitel, “Java How to Program”, Eleventh Edition, Deitel & Associates, Inc.l , 2018.
4. Timothy Budd, “Understanding Object Oriented Programming with Java”,Updated edition, Pearson Education, 2013.
5. Kathy Sierra & Bert Bates, ”Head First Java”, 2nd Edition, Oreilly.

Web Resources

1. **NPTEL**, ”Programming in Java,” Computer Science and Engineering, [Online]. Available: `https://archive.nptel.ac.in/courses/106/105/106105191/`. [Accessed: Feb. 25, 2025].
2. **NPTEL**, ”Data Structure and Algorithms using Java,” Computer Science and Engineering, [Online]. Available: `https://onlinecourses.nptel.ac.in/noc24_cs96/preview`. [Accessed: Feb. 25, 2025].
3. **Coursera**, ”Java Programming and Software Engineering Fundamentals,” [Online]. Available: `https://www.coursera.org/specializations/java-programming`. [Accessed: Feb. 25, 2025].
4. **Coursera**, ”Programming in Java: A Hands-on Introduction Specialization,” [Online]. Available: `https://www.coursera.org/specializations/hands-on-java`. [Accessed: Feb. 25, 2025].

COURSE CODE: 24CS203
SOFTWARE ENGINEERING

Course Category:	Professional Core (PC)	**Credits:**	3
Course Type:	Theory	**Lecture -Tutorial-Practice:**	3-0-0
Pre-requisites:	—	**Continuous Evaluation:** **Semester end Evaluation:** **Total Marks:**	40 60 100

Course Description

This course provides an overview of the software engineering discipline, introducing the student to the fundamental principles and methods in software engineering. The course teaches the student various-methods and models to gather the system requirements, analyze and model the specified requirements both functional and non-functional requirements including system constraints. The course also aims at teaching various existing design models and patterns for different domains. Design methodologies including structured design and object-oriented design are also covered. Students will learn the implementation and testing concepts in developing a successful software solution that meets the specified requirements.

Course Objectives

- To understand the Nature of Software
- To design Class-Based and Component-Level Models
- To evaluate and Implement Testing Strategies for Different Software Types
- To apply Software Engineering Knowledge to implement Real-World Projects
- To apply various software testing strategies.

Course Outcomes

At the end of the course, the student will be able to

	Course Outcomes	BTL	POI
CO1	Understand the basics of software engineering.	**K2**	1.7.1,2.5.1, 2.5.2, 2.6.4, 2.7.1
CO2	Understand the different software engineering process models.	**K2**	1.7.1,2.5.1, 2.5.2, 2.6.4, 2.7.1
CO3	Apply analysis model for any given application.	**K3**	1.7.1,2.5.1,2.7.1,3.5.1, 3.5.2,3.5.6
CO4	Apply design model for any given application.	**K3**	1.7.1,2.5.1,2.6.4 3.5.1,3.8.1
CO5	Apply different testing techniques.	**K3**	1.7.1,2.5.1,2.6.4, 3.5.1,3.6.2,3.8.2

Contribution of Course Outcomes towards achievement of Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2
CO1	2	3											2	2
CO2	2	3											2	2
CO3	2	2	3										1	2
CO4	2	2	3										1	2
CO5	2	2	3										1	2

Course Content

Unit – I : Process Models and Agile Development

Introduction: The Nature of Software, The changing nature of software, the Software Process, Software Engineering Practice, Software Development Myths.
Process models: Prescriptive Process Models, Specialized Process Models, The Unified Process, Personal and Team Process Models.
Agile Development: What Is Agility? Agility and the Cost of Change. What Is an Agile Process? Extreme Programming (XP) Other Agile Process Models, A Tool Set for the Agile Process.

Unit – II : Requirements Engineering

Understanding Requirements: Requirements Engineering, Establishing the Groundwork, Eliciting Requirements, Developing Use Cases, Building the Analysis Model, Negotiating Requirements, Requirements monitoring, and Validating Requirements
Requirements Modeling(Scenarios, Analysis Classes): Scenario Based Methods: Requirements Analysis, Scenario-Based Modeling, UML Models That Supplement the Use Case, Class based methods, Identifying Analysis classes, Specifying attributes, Defining operators, Class-Responsibility-Collaborator Modeling, Associates and Dependencies, Analysis Packages

Unit – III : Requirements Modeling & Design Concepts

Requirements Modeling(Flow and Behavior): Behavior, Patterns, And Web apps: Creating a Behavioral Model, Identifying events with Use Cases, State Representations, Patterns for Requirements Modeling, Requirements Modeling for Web and Mobile Apps
Design Concepts: Design within the Context of Software Engineering, the Design Process, Design Concepts, the Design Model.

Unit – IV : Software Design

Architectural Design: Software Architecture, Architectural Genres, Architectural Styles, Architectural Considerations, Architectural Decisions, Architectural Design.
Component-Level Design: What Is a Component? Designing Class-Based Components, Conducting Component Level Design, and Component level design for Web Apps, Component Level Design for Mobile Apps.
User Interface Design: The Golden Rules, User Interface Analysis and Design, Interface Analysis, Interface Design Steps, Web app and Mobile Interface Design.

Unit – V : Software Testing

Software Testing Strategies: A Strategic Approach to Software Testing, Strategic Issues, Test Strategies for Conventional Software, Test Strategies for Object-Oriented Software,
Testing Conventional Applications: Software Testing Fundamentals, Internal and External Views of Testing, White Box Testing, Basis Path Testing, Control Structure Testing, Black-Box Testing

Text Books

1. Roger S. Pressman, Bruce R.Maxim, "Software Engineering a practitioners approach" 8th edition, McGraw-Hill Publication, 2019.

Reference Books

1. Ian Somerville, "Software Engineering". 9th ed, Pearson Education. 2011.
2. Carlo Ghezzi, Mehdi Jazayeri and Dino Mandrioli, "Fundamentals of Software Engineering".2 ed, PHI. 2009
3. RajibMall, Fundamentals of Software Engineering. 3 ed, PHI. 2009.

Web Resources

1. **NPTEL**, "Lecture Series on Software Engineering," Department of Computer Science & Engineering, IIT Bombay. [Online]. Available:`https://nptel.ac.in/courses/106101061/2`. [Accessed: Mar. 9, 2025].
2. **NPTEL**, "Software Engineering," by Dr. B. Lavanya, Assistant Professor, University of Madras. [Online]. Available: `https://onlinecourses.swayam2.ac.in/cec20_cs07/preview`. [Accessed: Mar. 9, 2025].
3. **NPTEL**, "Software Engineering Basics," [Online]. Available: `https://www.youtube.com/watch?v=sB2iQSvrcG0`. [Accessed: Mar. 9, 2025].

COURSE CODE: 24CS204
OPERATING SYSTEMS

Course Category:	Program Core (PC)	**Credits:**	4
Course Type:	Theory	**Lecture -Tutorial-Practice:**	3-0-2
Pre-requisites:	Programming using 'C'	**Continuous Evaluation:** **Semester end Evaluation:** **Total Marks:**	40 60 100

Course Description

This course provides a comprehensive introduction to understand the underlying principles, techniques and approaches which constitute a coherent body of knowledge in operating systems. In particular, the course will consider inherent functionality and processing of program execution. The emphasis of the course will be placed on understanding how the various elements that constitute an operating system interact and provides services for execution of application software.

Course Objectives

- To understand OS structures and process management, including system calls, scheduling, and interprocess communication.
- To analyze CPU scheduling and synchronization, covering scheduling algorithms, thread management, and classic synchronization problems.
- To comprehend deadlock concepts and handling, focusing on prevention, avoidance, detection, and recovery strategies.
- To explore memory management techniques, including address binding, paging, swapping, and fragmentation handling.
- To apply OS concepts to practical scenarios, implementing process management, scheduling, synchronization, deadlocks, and memory allocation.

Course Outcomes

At the end of the course, the student will be able to

	Course Outcomes	BTL	POI
CO1	Identify the basic components of an Operating System engineering.	**K2**	1.7.1,2.5.1, 2.5.2, 2.6.4, 2.7.1
CO2	Apply CPU Scheduling and disk scheduling algorithms to achieve specific criteria	**K3**	1.7.1,2.5.1, 2.5.2, 2.6.4, 2.7.1
CO3	Apply analysis model for any given application.	**K3**	1.7.1,2.5.1,2.7.1,3.5.1, 3.5.2,3.5.6
CO4	Analyze the mechanisms used for process synchronization and handling deadlocks	**K4**	1.7.1,2.5.1,2.6.4 3.5.1,3.8.1
CO5	Analyze virtual memory techniques & File system Implementation techniques	**K4**	1.7.1,2.5.1,2.6.4, 3.5.1,3.6.2,3.8.2

Contribution of Course Outcomes towards achievement of Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2
CO1	2													
CO2	1	2	1										1	
CO3	1	2	2										1	
CO4	1	2	2										1	
CO5	1	2	2										1	

Course Content

Unit – I : Introduction and Operating System Structures

Introduction: Operating Systems, Computer-System Organization, Computer-System Architecture, Operating-System Operations, Distributed Systems, Free and Open-Source Operating Systems.
Operating-System Structures: Operating-System Services, User and Operating-System Interface, System Calls, System Services, Operating-System Design and Implementation, Operating-System Structure, Building and Booting an Operating System.

Unit – II : Process Management

Processes and Threads: Process Concept, Process Scheduling, Operations on Processes, Interprocess Communication, IPC in Shared-Memory Systems, IPC in Message-Passing Systems, Threads Overview, Multithreading Models, Pthreads, Threading Issues.
CPU Scheduling: Basic Concepts, Scheduling Criteria, Scheduling Algorithms (FCFS, SJF, RR, Priority, Multilevel Queue, Multilevel Feedback Queue), Multi-Processor Scheduling.

Unit – III : Process Synchronization

Synchronization Tools: Background, The Critical-Section Problem, Peterson's Solution, Hardware Support for Synchronization, Mutex Locks, Semaphores, Monitors, Classic Problems of Synchronization.
Deadlocks: System Model, Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery from Deadlock.

Unit – IV : Memory Management

Main Memory: Background, Contiguous Memory Allocation, Paging, Structure of the Page Table, Swapping.
Virtual Memory: Background, Demand Paging, Copy-on-Write, Page Replacement, Allocation of Frames, Thrashing.

Unit – V : Storage Management and File System

Mass-Storage Structure: Overview of Mass-Storage Structure, HDD Scheduling (FCFS, SCAN, C-SCAN), RAID Structure.
File-System Interface: File Concept, Access Methods, Directory Structure
File-System Implementation: File-System Structure, File-System Operations, Directory Implementation, Allocation Methods.
File-System Internals: File Systems, File-System Mounting, Partitions and Mounting, File Sharing

List of Programming Tasks

1. Introduction to Linux Command Line Interface and Shell Utilities.
2. Practicing Essential Windows PowerShell Commands Using Batch Scripts.
3. Implementation of Fundamental UNIX System Calls for Process and File System Management.
4. Simulation of FCFS, SJF (Preemptive & Non-Preemptive), Round Robin Scheduling Algorithms.
5. Simulation of Priority Scheduling (Preemptive & Non-Preemptive.
6. Real-Time Scheduling – EDF and Rate Monotonic Scheduling (RMS).
7. Simulating the Critical Section Problem & Implementing Peterson's Solution.
8. Implementing Producer-Consumer Problem using Semaphores.
9. Solving Readers-writers problem using synchronization tools.
10. Solving the Dining Philosophers Problem.
11. Simulate Banker's Algorithm for Deadlock Avoidance.
12. Simulate memory allocation strategies like First Fit, Best Fit, and Worst Fit.
13. Simulate logical to physical address translation using fixed-size pages and frames.
14. Implement and compare FIFO, LRU, and Optimal page replacement algorithms.
15. Simulate disk scheduling algorithms like FCFS, SSTF, SCAN, C-SCAN to calculate total head movement.
16. Simulate and compare File Allocation Methods including Contiguous Allocation, Linked Allocation & Indexed Allocation.
17. Implement free space tracking methods: Bit vector, Linked list & Grouping.

Text Books

1. Abraham Silberschatz, Peter B. Galvin and Greg Gagne, "Operating SystemConcepts", 10th Edition, John Wiley & Sons Pvt. Ltd, 2018.

Reference Books

1. William Stallings, "Operating System: Internals and Design Principles", 6thEdition 2009.
2. Andrew S.Tanenbaum, "Modern Operating Systems", 3rdEdition, PHI, 2008.

Web Resources

1. **NPTEL**, "Lecture Series on Operating Systems," Department of Computer Science & Engineering, IIT Kharagpur. [Online]. Available: `https://nptel.ac.in/courses/106/105/106105214/`. [Accessed: Mar. 9, 2025].

2. **Stanford University**, "Lecture Notes on Operating Systems," Department of Computer Science. [Online]. Available: `https://www.scs.stanford.edu/21wi-cs140/notes/`. [Accessed: Mar. 9, 2025].

3. **IIT Bombay**, "Lecture Notes on Operating Systems," Department of Computer Science & Engineering. [Online]. Available: `https://www.cse.iitb.ac.in/~mythili/os/`. [Accessed: Mar. 9, 2025].

COURSE CODE: 24CS281
OBJECT ORIENTED PROGRAMMING THROUGH JAVA LAB

Course Category:	Program Core (PC)	**Credits:**	1.5
Course Type:	Practical	**Lecture -Tutorial-Practice:**	0-0-3
Pre-requisites:	Programming using 'C' Lab	**Continuous Evaluation:** **Semester end Evaluation:** **Total Marks:**	60 40 100

Course Description

The course provides hands-on experience in developing applications using Java, focusing on core and advanced object-oriented programming concepts. The course begins with the installation and basic setup of the Java Development Kit (JDK) and guides students through writing, compiling, and executing simple Java programs. It introduces control statements, primitive data types, arrays, type conversions, and object-oriented principles such as classes, objects, constructors, method overloading, and string handling. Students will gain practical knowledge of inheritance, method overriding, dynamic method dispatch, exception handling, and modular programming using packages and interfaces. The course also covers file handling through byte and character streams. In the advanced part of the lab, students will explore multithreading, thread priorities, lambda expressions, and generic functional interfaces. The course concludes with the implementation of the Java Collections Framework and the Stream API for processing data effectively. This lab course is designed to strengthen students' programming skills, improve problem-solving abilities, and prepare them to develop scalable, maintainable Java applications in real-world scenarios.

Course Objectives

- To familiarize with the installation and basic structure of Java programs, including primitive data types and control structures.
- To introduce and develop understanding of arrays, object-oriented programming concepts such as classes, objects, constructors, method overloading, inheritance, method overriding, and dynamic method dispatch.
- To equip with the knowledge of modular programming using packages, interfaces, and exception handling mechanisms.
- To develop applications with the concepts of multithreading and lambda expressions.
- To provide hands-on experience with generic functional interfaces, and Java collection framework including Stream API.

Course Outcomes

At the end of the course, the student will be able to

	Course Outcomes	BTL	POI
CO1	Apply the basic concepts of Java including installation, data types, control statements, and arrays.	**K3**	1.7.1, 2.6.3, 3.5.1
CO2	Apply OOP principles, string handling and inheritance.	**K3**	1.7.1, 2.5.2, 3.5.1, 5.4.1
CO3	Apply the concepts of packages, interfaces, exception handling, and stream-based file input/output.	**K3**	1.7.1, 2.5.2, 3.5.1, 5.4.1
CO4	Apply the concepts of multithreading and lambda expressions.	**K3**	1.7.1, 2.7.1, 3.7.1, 5.4.2
CO5	Apply the concepts of Java collections framework and Stream API.	**K3**	1.7.1, 2.7.1, 3.7.1, 5.4.1

Contribution of Course Outcomes towards achievement of Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2
CO1	2	3	1										1	
CO2	2	3	1										2	2
CO3	1	2	2		3								2	2
CO4	1	2	3		2								2	2
CO5	2	2	1		3								2	2

Course Content

Task 1: Installation, Simple Program and Control Statements

- **Program 1:** Installation of Java and Execution of a Simple Hello World Program.
- **Program 2:** Implement a program to demonstrate the usage two control statements (if statement and for loop).

Task 2: Primitive Types, Type Conversions and Type Promotions

- **Program 1:** Implement a program to demonstrate the eight primitive types of data.
- **Program 2:** Implement a program to demonstrate type conversions and type promotions

Task 3: Arrays

- **Program 1:** Implement a program to demonstrate a two-dimensional array.
- **Program 2:** Implement a program by applying the concepts of arrays for a given use case.

Task 4: Classes, Objects and Constructors

- **Program 1:** Implement a program to demonstrate classes and objects.
- **Program 2:** Implement a program to demonstrate constructors and this keyword.

Task 5: Method Overloading and String Handling

- **Program 1:** Implement a program to demonstrate method overloading.
- **Program 2:** Implement a program to demonstrate String, StringBuffer and StringTokenizer class.

Task 6: Inheritance, Method Overriding and Dynamic Method Dispatch

- **Program 1:** Implement a program to demonstrate different types of inheritance and super keyword.
- **Program 2:** Implement a program to demonstrate method overriding and dynamic method dispatch. .

Task 7: Packages, Interfaces and Exceptions

- **Program 1:** Implement a program to demonstrate packages and interfaces.
- **Program 2:** Implement a program to demonstrate built-in exceptions and custom exception.

Task 8: Byte Streams and Character Streams

- **Program 1:** Implement a program to demonstrate byte streams.
- **Program 2:** Implement a program to demonstrate character streams.

Task 9: Threads, Multiple Threads and Thread Priorities

- **Program 1:** Implement a program to demonstrate creation of threads and multiple threads.
- **Program 2:** Implement a program to demonstrate thread priorities.

Task 10: Generic Functional Interfaces and Lambda Expressions

- **Program 1:** Implement a program to demonstrate generic functional interfaces.
- **Program 2:** Implement a program to demonstrate passing lambda expressions as arguments.

Task 11: Collection Classes

- **Program 1:** Implement a program to demonstrate ArrayList Class, LinkedList Class, HashSet Class.
- **Program 2:** Implement a program to demonstrate HashMap Class, TreeMap Class.

Task 12:The Stream API

- **Program 1:** Implement a program to demonstrate reduce operations in Stream API.
- **Program 2:** Implement a program to demonstrate iterators and streams in Stream API.

Text Books

1. Herbert Schildt, Danny Coward, “Java: The Complete Reference”, Thirteenth Edition, McGrawHill, 2023.

Reference Books

1. Herbert Schildt, Dale Skrien, “Java Fundamentals A Comprehension Introduction”, Special Indian Edition, McGraw-Hill Education India Pvt. Ltd, 2017.
2. E Balaguruswamy, ”Programming with Java”, Seventh Edition, Mc Graw Hill 2023.
3. Paul J. Dietel and Dr.Harvey M. Deitel, “Java How to Program”, Eleventh Edition, Deitel & Associates, Inc.l , 2018.
4. Timothy Budd, “Understanding Object Oriented Programming with Java”,Updated edition, Pearson Education, 2013.
5. Kathy Sierra & Bert Bates, ”Head First Java”, 2nd Edition, Oreilly.

Web Resources

1. **NPTEL**, "Programming in Java," Computer Science and Engineering, [Online]. Available: `https://archive.nptel.ac.in/courses/106/105/106105191/`. [Accessed: Feb. 25, 2025].
2. **NPTEL**, "Data Structure and Algorithms using Java," Computer Science and Engineering, [Online]. Available: `https://onlinecourses.nptel.ac.in/noc24_cs96/preview`. [Accessed: Feb. 25, 2025].
3. **Coursera**, "Java Programming and Software Engineering Fundamentals," [Online]. Available: `https://www.coursera.org/specializations/java-programming`. [Accessed: Feb. 25, 2025].
4. **Coursera**, "Programming in Java: A Hands-on Introduction Specialization," [Online]. Available: `https://www.coursera.org/specializations/hands-on-java`. [Accessed: Feb. 25, 2025].

COURSE CODE: 24CS282
DIGITAL LOGIC DESIGN LAB

Course Category:	Engineering Sciences (ES)	**Credits:**	1.5
Course Type:	Theory	**Lecture -Tutorial-Practice:**	0-0-3
Pre-requisites:	—	**Continuous Evaluation:** **Semester end Evaluation:** **Total Marks:**	60 40 100

Course Description

This laboratory component is designed to help students gain practical experience in implementing digital circuits and systems based on the theory learned in the classroom. The experiments cover a wide range of topics, from basic logic gates to more complex combinational and sequential circuits, and also introduce the use of FPGA for digital design.

Course Objectives

- To understand and verify the functionality of logic gates and their realization using universal gates.
- To design and implement basic arithmetic circuits, code converters, and comparators
- To implement combinational using multiplexers, decoders, and encoders.
- To develop and analyze sequential circuits, including flip-flops, counters, and shift registers.
- To introduce FPGA-based logic design and implement digital circuits using Xilinx Spartan-6.

Course Outcomes

At the end of the course, the student will be able to

	Course Outcomes	BTL	POI
CO1	Apply knowledge of logic gates and universal gates for circuit realization.	**K3**	1.2.1, 1.7.1
CO2	Apply combinational logic concepts to design arithmetic circuits, code converters, and comparators.	**K3**	1.2.1, 1.7.1, 2.7.2
CO3	Analyze combinational circuits using MSIs such as multiplexers, decoders, and encoders.	**K4**	1.7.1, 2.7.1, 2.7.2, 3.7.1 ,3.8.1
CO4	Analyze various types of flip-flops, counters, and shift registers.	**K4**	1.7.1, 2.7.1, 2.7.2, 3.7.1 ,3.8.1
CO5	Analyze logic gate realization and combinational circuit design using FPGA platforms.	**K4**	1.7.1, 2.7.1, 2.7.2, 3.7.1 ,3.8.1

Contribution of Course Outcomes towards achievement of Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2
CO1	2													
CO2	2	2												
CO3	2	2	2											1
CO4	2	2	2											1
CO5	1	2	2											

Course Content

Task 1: Realization of All logic gates using universal gates.

- Verification of logic gates from DIGITAL IC'S.
- Realization of logic gates using NAND and NOR.

Task 2: Design and Implementation of Arithmetic circuits.

- Design and Implementation of half adder and half sub tractor.
- Design and Implementation of half adder and half sub tractor.

Task 3: Design and implement different types of code converters

- Design and implement i) Binary to Gray ii) Gray to Binary code converters
- Design and implement i) BCD to EX-3 ii) EX-3 to BCD code converters.

Task 4: Design and implementation of magnitude comparators

- Design and implement single bit comparator.
- Design and implement single bit comparator.

Task 5: Implementation of Decoders and encoders

- Implementation of i) 2X4 Decoder ii)3X8 Decoder iii) BCD to Decimal Decoder iv) BCD to 7 segment Display
- Implementation of i) 4x2 Encoder ii) Octal to Binary encoder iii) Decimal to BCD encoder.

Task 6: Implementation of Multiplexer and De Multiplexer.

- Implementation of i) 2X1 MUX ii)4X1 MUX
- Implementation of i) 1X2 De MUX ii)1X4 De MUX

Task 7: Implementation of all types of FLIP-FLOPS using gates.

- Implementation of SR latch using NAND & NOR
- Implementation of SR, JK, D, T flip flops.
- Implementation of Master-Slave JK Flip flop

Task 8: Design of Synchronous counters

- Design and implementation of synchronous up, synchronous down counter and Up-down Counters .
- Design and implementation of synchronous MOD counters.

Task 9: Design of Asynchronous counters.

- Design of Asynchronous up and down counters.
- Design and implementation of Asynchronous Mod counters.

Task 10: Design of Ring-counter and Johnson counter.

- Design and implementation of 4-bit Ring counter.
- Design and implementation of 4-bit Johnson counter.

Task 11: Verification of Shift-Registers using flip flops.

- Verification of i) SISO ii) SIPO Shift registers
- Verification of i) PISO ii) PIPO Shift registers

Task 12: Logic Design with FPGA

- Introduction to Xilinx Spartan-6 FPGA Development Board.
- realization of logic gates
- Realization of combinational logic circuits.

Text Books

1. M.Morris Mano, Digital Logic & Computer Design 1 e/d reprint, Pearson education, 2016
2. M.Morris Mano, Michael D Ciletti Digital Design with an Introduction toVerilog HDL 5th e/d, Pearson education, 2013

Reference Books

1. A. Anand Kumar, Switching Theory and Logic Design, 2nd Edition, PHI,2013
2. Charles H.Roth , Fundamentals of Logic Design, 6/e, Cengage learning,2010
3. Computer Architecture and Organization Designing for Performance, William Stallings, Ninth edition, Pearson Education series, 2014.

Web Resources

1. **NPTEL**, "Digital Systems," Prof. N. Goel, Department of Electrical Engineering, IIT Ropar, [Online]. Available: `https://nptel.ac.in/courses/108/106/108106177/`. [Accessed: Feb. 25, 2025].
2. **NPTEL**, "Digital Systems," Prof. N. J. Rao, Department of Electrical Engineering, IISc Bangalore, NPTELWEB Notes, [Online]. Available: `https://nptel.ac.in/courses/106/108/106108099/`. [Accessed: Feb. 25, 2025].

COURSE CODE: 24UC201
UNIVERSAL HUMAN VALUES-II

Course Category:	Mandatory Course (MC)	**Credits:**	3
Course Type:	Theory	**Lecture -Tutorial-Practice:**	2-1-0
Pre-requisites:	UHV-I (Student Induction Programme)	**Continuous Evaluation:** **Semester end Evaluation:** **Total Marks:**	40 60 100

Course Description

The course is mandated by AICTE for all B.Tech. students of all branches, preferably in the second year. It is intended to facilitate the development of a holistic and humane world vision. This course employs an innovative and effective methodology of self-exploration, self-verification on one's own right. It is presented as a systematic set of universal, rational and verifiable proposals about human reality, about the inherent harmony in the human being, the family, society, the entire nature and existence. It draws out the value or role of human being in the harmony at all these levels, which is essentially the scope of ethical human conduct. It further helps to find resolution of present-day challenges. The issues in professional ethics are analyzed in the context of the understanding of harmony. While handling the course, the teacher is also a co-explorer along with the students.

Course Objectives

- To help the students appreciate the essential complementarily between 'values' and 'skills' to ensure sustained happiness and prosperity which are the core aspirations of all human beings.
- To facilitate the development of a holistic perspective among learners towards life and profession.
- To facilitate the right understanding of happiness and prosperity based on correct understanding of human reality and the rest of existence.
- To identify that a holistic perspective forms the basis of universal human values and movement towards value-based living in a natural way.
- To highlight plausible implications of holistic understanding in terms of ethical human conduct, trustful and mutually fulfilling human behavior and mutually enriching interaction with Nature.

Course Outcomes

At the end of the course, the student will be able to

	Course Outcomes	BTL	POI
CO1	Apply the right understanding of the concepts of value education and basic human aspirations through self-exploration for the fulfillment of human aspirations.	**K3**	6.3.1, 6.4.1, 7.3.1, 8.4.1, 9.5.2, 9.5.3, 10.5.1, 12.5.2
CO2	Analyse various aspects of the human being as the combination of Self and Body for attaining harmony at the level of human being (individual)	**K4**	6.3.1, 6.4.1, 7.3.1, 8.4.1, 9.5.2, 9.5.3
CO3	Apply the knowledge of nine universal values in human-human relationship for harmony at the level of family, and appreciate all the essential factors that help in attaining harmony at the level of society.	**K3**	6.3.1, 6.4.1, 7.3.1, 8.4.1, 9.5.2, 9.5.3, 10.5.1, 12.5.2
CO4	Differentiate the characteristics and activities of various orders of Nature and study the mutual fulfillment among them, and also identify the existence as co-existence at all levels	**K4**	6.3.1, 6.4.1, 7.3.1, 8.4.1, 9.5.2, 9.5.3, 10.5.1, 12.5.2
CO5	Present sustainable solutions to various challenges in society and Nature, and identify that the solutions are practicable	**K3**	6.3.1, 6.4.1, 7.3.1, 8.4.1, 9.5.2, 9.5.3

Contribution of Course Outcomes towards achievement of Program Outcomes

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2
CO1						3						3		
CO2						3						3		
CO3						3			2	2		3		
CO4						3	3		2	2		3		
CO5						3	3	3	3	3		3		

Course Content

Unit – I :Introduction, Need, and Basic Guidelines

Purpose and Motivation: Recapitulation from UHV-I, Self-exploration: what is it?, its content and process, 'Natural acceptance' and experimental validation – as the process for self-exploration, Continuous happiness and prosperity – a look at basic human aspirations. Right understanding, relationship and physical facility – the basic requirements for fulfillment of aspirations of every human being with their correct priority, understanding happiness and prosperity correctly – a critical appraisal of the current scenario, method to fulfill the above human aspirations: understanding and living in harmony at various levels.

Examples/Applications/Case Studies:

1. Examples to differentiate the meaning of happiness and prosperity in the current scenario and the meaning proposed in the course.
2. Case studies related to the basic aspirations assumed by the students and the real basic aspirations.

Exercises/Projects/Practices: Practice sessions are to be included to discuss natural acceptance in human being as the innate acceptance for living with responsibility (living in relationship, harmony and co-existence) rather than as arbitrariness in choice based on liking-disliking.

Unit – II: Understanding Harmony in the Human Being – Harmony in Myself

Understanding Human Being: Understanding human being as a co-existence of the sentient 'I' and the material 'Body'. Understanding the needs of Self ('I') and 'Body' – happiness and physical facility, Understanding the Body as an instrument of 'I' (I being the doer, seer and enjoyer). Understanding the characteristics and activities of 'I' and harmony in 'I'. Understanding the harmony of I with the Body: Sanyam and Health; correct appraisal of Physical needs, meaning of Prosperity in detail, Programs to ensure Sanyam and Health.
Examples/Applications/Case Studies:

1. Examples to differentiate the functions and needs of both Self and Body.
2. Case studies related to the Sanyam and Health and correct appraisal of physical needs that provides clarity on the real meaning of prosperity.

Exercises/Projects/Practices: Practice sessions are to be included to discuss the role others have played in making material goods available to me. Identifying from one's own life. Differentiate between prosperity and accumulation. Discuss program for ensuring health vs. dealing with disease.

Unit – III: Understanding Harmony in the Family and Society – Harmony in Human-Human Relationship

Understanding Values: Understanding values in human-human relationship; meaning of Justice (nine universal values in relationships) and program for its fulfillment to ensure mutual happiness; Trust and Respect as the foundational values of relationship, Understanding the meaning of Trust; Difference between intention and competence, Understanding the meaning of Respect, Difference between respect and differentiation; the other salient values in relationship. Understanding the harmony in the society (society being an extension of family); Resolution, Prosperity, fearlessness (trust) and co-existence as comprehensive Human Goals, Visualizing a universal harmonious order in society–Undivided Society, Universal Order–from family to world family.
Examples/Applications/Case Studies:

1. Examples for exploring the nine values in human-human relationships in family.
2. Case studies for visualizing universal harmonious order in society and universal order.

Exercises/Projects/Practices: Practice sessions are to be included to reflect on relationships in family, hostel and institute as extended family, real life examples, teacher-student relationship, goal of education, etc. Gratitude as a universal value in relationships. Discuss with scenarios. Elicit examples from students' lives.

Unit – IV :Understanding Harmony in Nature & Existence – Whole existence as Coexistence

Understanding the Nature Harmony: Understanding the harmony in the Nature, Interconnectedness and mutual fulfillment among the four orders of Nature – recyclability and self-regulation in nature, Understanding Existence as Co-existence of mutually interacting units in all-pervasive space, Holistic perception of harmony at all levels of existence.
Examples/Applications/Case Studies:

1. Examples to distinguish various units in all the four orders of Nature.
2. Case studies related to recyclability and self-regulation in Nature.

Exercises/Projects/Practices: Practice sessions are to be included to discuss human being as cause of imbalance in nature (film "Home" can be used), pollution, depletion of resources and role of technology, etc.

Unit – V: Implications of the above Holistic Understanding of Harmony on Professional Ethics

Human Values: Natural acceptance of human values, Definitiveness of ethical human conduct, Basis for humanistic education, humanistic constitution and humanistic universal order, Competence in professional ethics: a) ability to utilize the professional competence for augmenting universal human order, b) ability to identify the scope and characteristics of people-friendly and eco-friendly production systems, c) ability to identify and develop appropriate technologies and management patterns for above production systems, Case studies of typical holistic technologies, management models and production systems, Strategy for transition from the present state to Universal Human Order: a) at the level of individual: as socially and ecologically responsible engineers, technologists and managers, b) at the level of society: as mutually enriching institutions and organizations.

Examples/Applications/Case Studies:

1. Examples for holistic technologies, management models and production systems.
2. Case studies related to production systems, management models and technologies.

Exercises/Projects/Practices: Practice exercises and case studies are to be taken up in practice (tutorial) sessions eg. to discuss the conduct as an engineer or scientist, etc.

Text Books

1. R.R. Gaur, R. Sangal, G.P. Bagaria, A Foundation Course in Human Values and Professional Ethics, Excel Books Private Limited, New Delhi, 2010.
2. R.R. Gaur, R. Asthana, G.P. Bagaria, A Foundation Course in Human Values and Professional Ethics (2nd revised ed.), Excel Books Private Limited, New Delhi, 2019.

Reference Books

1. A. Nagaraj, Jeevan Vidya: Ek Parichaya, Jeevan Vidya Prakashan, Amarkantak, 1999.
2. A.N. Tripathi, Human Values, New Age International Publishers, New Delhi, 2004.
3. Annie Leonard, The Story of Stuff: The Impact of Overconsumption on the Planet, Our Communities, and Our Health and How We Can Make It Better, Free Press, New York, 2010.
4. Mohandas Karamchand Gandhi, The Story of My Experiments with Truth: Mahatma Gandhi Autobiography, B.N. Publishing, 2008.
5. E.F. Schumacher, Small is Beautiful: A Study of Economics as if People Mattered, Vintage Books, London, 1993.
6. Cecile Andrews, Slow is Beautiful: New Visions of Community, New Society Publishers, Canada, 2006.
7. J.C. Kumarappa, Economy of Permanence, Sarva-Seva-Sangh Prakashan, Varanasi, 2017.
8. Pandit Sunderlal, Bharat Mein Angreji Raj, Prabhath Prakashan, Delhi, 2018.
9. Dharampal, Rediscovering India, Society for Integrated Development of Himalayas, 2003.
10. M.K. Gandhi, Hind Swaraj or Indian Home Rule, Navajivan Publishing House, Ahmedabad, 1909.
11. Maulana Abul Kalam Azad, India Wins Freedom: The Complete Version, Orient Blackswan, 1988.
12. Romain Rolland, The Life of Vivekananda and the Universal Gospel, Advaita Ashrama, India, 2010.

13. Romain Rolland, Mahatma Gandhi: The Man Who Became One with the Universal Being, Srishti Publishers & Distributors, New Delhi, 2002.

Web Resources

1. R.R. Gaur, R. Sangal, G.P. Bagaria, **A Foundation Course in Human Values and Professional Ethics**, [Online]. Available: `https://dokumen.pub/a-foundation-course-in-human\-values-and-professional-ethics-firstnbsped-9788174467812.html`. [Accessed: Jun. 13, 2025].

2. **AICTE**, "AICTE – SIP YouTube Channel," [Online]. Available: `https://www.youtube.com/channel/UCo8MpJB_aaVwB4LWLAx6AhQ`. [Accessed: Jun. 13, 2025].

3. **AICTE**, "AICTE – UHV Teaching Learning Material," [Online]. Available: `https://fdp-si.aicte-india.org/download.php#1`. [Accessed: Jun. 13, 2025].