Vidya Pratishthan's Kamalnayan Bajaj Institute of Engineering and Technology

Vidyanagari, Baramati, Dist. – Pune 413133
An Autonomous Institute Approved by AICTE and affiliated to SPPU,Pune

Department of Computer Engineering



Curriculum Structure and Syllabus of Second Year B. Tech Computer Engineering (Course 2025)

With effective from Academic Year 2026-27

INSTITUTE VISION AND MISSION

VISION

To achieve Academic Excellence through Persistent and Synergic Collaborations amongst all Stakeholders.

MISSION

1. To ensure holistic development of students as lifelong learners and problem solvers through value based quality education.

2. To motivate faculty to attain the state-of-the-art knowledge and wisdom in their domain and be a facilitator towards co- creation of knowledge

3. To frame and deploy conducive and empowering policies for multifaceted growth of students, faculty and staff to make them contributors towards excellence.

4. To partner with industry for mutually beneficial relations to generate employable and deployable workforce.

5. To fulfill the aspirations of alumni, parents, society, region and nation at large by generating technically competent.

DEPARTMENT VISION AND MISSION

VISION

To achieve excellence in the field of Computer Engineering with consistent and collaborative efforts of every individual

MISSION

1. To develop students with fundamental advanced tools and technologies to work as skilled Computer professionals with ethical values.

2. To promote faculty for higher education and expose them to current trends to enrich educational quality.

3. To provide appropriate environment with required resources to achieve academic excellence.

4. To develop hand-in-hand relations with industries for catering institute-industry needs.

5. To apply collaborative efforts to make students competent to provide solutions to social problems.

Program Specific Outcomes (PSO)

PSO1: Professional Skills

- Apply knowledge of algorithms, web design, databases, operating systems, data analytics, and networking to understand, analyze, and develop computer programs for the efficient design of computer-based systems with varying levels of complexity

PSO2: Problem-Solving Skills

- The capability to implement standard practices and strategies in software project development within open-ended programming environments, capability to utilize modern programming languages, environments, and platforms to build innovative career paths, pursue entrepreneurship, and foster enthusiasm for higher studies.

Program Educational Objectives (PEO)

1. Students will be able to apply the fundamentals, domain knowledge and modern technology of computer engineering to a provide effective and innovative solutions to engineering problems

2. Students will be able to solve societal challenging and multidisciplinary problems applying suitable resources

3. Students will be able to work as competent professional as an individual and a team member

Program Outcomes (POs)

PO1	***Engineering Knowledge:*** Apply knowledge of mathematics, natural science, computing, engineering fundamentals and an engineering specialization as specified in WK1 to WK4 respectively to develop to the solution of complex engineering problems.
PO2	***Problem Analysis:*** Identify, formulate, review research literature and analyze complex engineering problems reaching substantiated conclusions with consideration for sustainable development. (WK1 to WK4)
PO3	***Design/Development of Solutions:*** Design creative solutions for complex engineering problems and design/develop systems/components/processes to meet identified needs with consideration for the public health and safety, whole-life cost, net zero carbon, culture, society and environment as required. (WK5)
PO4	***Conduct Investigations of Complex Problems:*** Conduct investigations of complex engineering problems using research-based knowledge including design of experiments, modelling, analysis & interpretation of data to provide valid conclusions. (WK8).
PO5	***Engineering Tool Usage:*** Create, select and apply appropriate techniques, resources and modern engineering & IT tools, including prediction and modelling recognizing their limitations to solve complex engineering problems. (WK2 and WK6)
PO6	***The Engineer and The World:*** Analyze and evaluate societal and environmental aspects while solving complex engineering problems for its impact on sustainability with reference to economy, health, safety, legal framework, culture and environment. (WK1, WK5, and WK7).
PO7	***Ethics:*** Apply ethical principles and commit to professional ethics, human values, diversity and inclusion; adhere to national & international laws. (WK9)
PO8	***Individual and Collaborative Team work:*** Function effectively as an individual, and as a member or leader in diverse/multi-disciplinary teams.
PO9	***Communication:*** Communicate effectively and inclusively within the engineering community and society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations considering cultural, language, and learning differences
PO10	***Project Management and Finance:*** Apply knowledge and understanding of engineering management principles and economic decision-making and apply these to one's own work, as a member and leader in a team, and to manage projects and in multidisciplinary environments.
PO11	***Life-Long Learning:*** Recognize the need for, and have the preparation and ability for i) independent and life-long learning ii) adaptability to new and emerging technologies and iii) critical thinking in the broadest context of technological change. (WK8)

Vidya Pratishthan's
Kamalnayan Bajaj Institute of Engineering and Technology
Board of Studies: Computer Engineering
Syllabus: Second Year (SY B. Tech.) Computer Engineering
2025 Pattern w.e.f. AY:2026-2027

SEMESTER-III

Course Type	Course Code	Course Name	Teaching Scheme			Examination Scheme and Marks							Credits			
			TH	PR	TUT	CAA	ISE	ESE	TW	PR	OR	Total	TH	PR	TUT	Total
PCC	CO25201TH	Discrete Mathematics	3	-	-	10	30	60	-	-	-	100	3	-	-	3
PCC	CO25202TH	Object Oriented Paradigms	3	-	-	10	30	60	-	-	-	100	3	-	-	4
PCC	CO24202PR	Object Oriented Paradigms	-	2	-	-	-	-	-	30	-	30	-	1	-	
PCC	CO25203TH	Digital Electronics and Computer Organization	3	-	-	10	30	60	-	-	-	100	3	-	-	4
PCC	CO25203PR	Digital Electronics and Computer Organization	-	2	-	-	-	-	-	-	30	30		1	-	
MDM	MD250XXTH	Multi-disciplinary minor	3	-	-	10	30	60	-	-	-	100	3	-	-	4
MDM	MD250XXPR	Multi-disciplinary minor		2	-				30	-	-	30		1	-	
OE	OE250XXTH	Open Elective	2	-	-	10	-	60	-	-	-	70	2	-	-	2
HSS	HS25201TH	Public Specking and Aptitude	1		-	40	-	-		-		40	1	-	-	2
HSS	HS25201PR	Public Specking and Aptitude	-	2	-	-	-	-	-	-	30	30		1		
VSEC	CO25204TH	Data Analytics and Visualization using Python	1	-	-	40	-	-	-	-		40	1	-	-	2
VSEC	CO25204PR	Data Analytics and Visualization using Python	-	2	-	-	-	-	-	30	-	30	-	1	-	
	Total		16	10	-	130	120	300	30	60	60	700	16	5	0	21

List of Electives, Multi-Disciplinary Minor and Open Elective

Multidisciplinary Minor Courses (4 Credit)			
Course Code	**Course Name**	**Course Code**	**Course Name**
EL25051	Photovoltaic Technology & Solor Power Systems	ET25052	Drone Technology
Open Elective Courses			
Course Code	**Course Name**	**Course Code**	**Course Name**
OE25001	Digital Marketing		

Dr. M. D. Shelar
Academic Coordinator

Dr. A. M. Jagtap
Head of Department

28/03/2026
Dr. S. M. Bhosle
Dean Academics

Dr. A. H. Kolekar
Controller of Examination

¶

Dr. S. B. Lande
Principal
Vidya Pratishthan's
Kamalnayan Bajaj Institute of
Engineering & Technology, Baramati
Vidyanagari, Baramati-413133

VPKBIET BARAMATI
Affiliated to S.P.P.U. Pune Id No./PU/PN/Engg./152/2000
PUNE (MS.)

Vidya Pratishthan's
Kamalnayan Bajaj Institute of Engineering and Technology
Board of Studies: Computer Engineering
Syllabus: Second Year (SY B. Tech.) Computer Engineering
2025 Pattern w.e.f. AY: 2026-2027

SEMESTER-IV

Course Type	Course Code	Course Name	Teaching Scheme			Examination Scheme and Marks							Credits			
			TH	PR	TUT	CAA	ISE	ESE	TW	PR	OR	Total	TH	PR	TUT	Total
BS	BS25201TH	Advanced Mathematics for Computer Engineering	3	-	-	10	30	60	-	-	-	100	3	-	-	3
PCC	CO25211TH	Data Structures and Algorithms	3	-	-	10	30	60	-	-	-	100	3	-	-	4
PCC	CO25211PR	Data Structures and Algorithms	-	2	-	-	-	-	-	30	-	30	-	1	-	
PCC	CO25212TH	Microprocessor	3	-	-	10	30	60	-	-	-	100	3	-	-	4
PCC	CO25212PR	Microprocessor	-	2	-	-	-	-	-	-	30	30	-	1	-	
PCC	CO25213TH	Operating System	3	-	-	10	30	60	-	-	-	100	3	-	-	4
PCC	CO25213PR	Operating System	-	2	-	-	-	-	-	-	30	30	-	1	-	
MDM	MD250XXTH	Multi-disciplinary minor	2	-	-	10	-	60	-	-	-	70	2	-	-	3
MDM	MD250XXPR	Multi-disciplinary minor	-	2	-		-	-	30	-	-	30		1	-	
EEMC	CO25204PR	Community Engagement. Project	-	4	-	-	-	-	40	-	30	70	2	-	-	2
HSS	HS25211TH	Environmental Studies	2	-	-	10	-	60	-	-	-	70	2	-	-	2
	Total		16	12	1	70	120	360	70	30	90	730	16	6	0	22

List of Electives, Multi-Disciplinary Minor and Open Elective

Multidisciplinary Minor Courses (3 Credit)			
Course Code	Course Name	Course Code	Course Name
EL25052	Fundamentals of Solar Technology	ET25055	Fundamentals of Drone Technology

Dr. M. D. Shelar
Academic Coordinator

Dr. A. M. Jagtap
Head of Department

Dr. S. M. Bhosle
Dean Academics

28/03/2026

Dr. A. H. Kolekar
Controller of Examination

Dr. S. B. Lande
Principal

Principal
Vidya Pratishthan's
Kamalnayan Bajaj Institute of
Engineering & Technology, Baramati
Vidyanagari, Baramati-413133

Semester I

CO25201: Discrete Mathematics		
Teaching Scheme: TH: 03 Hrs/Week	***Credit: 03***	***Examination Scheme:*** Course Activity: 10 Mark In Semester: 30 Mark End Semester: 60 Mark

Prerequisite: Basic Mathematics

Course Objective:

To introduce students to understand, explain, and apply the foundational mathematical concepts at the core of computer science.

To apply appropriate set, function, and relation models to analyze practical examples.

To acquire skills in logic and proof techniques to enhance mathematical maturity.

To acquire a comprehensive understanding of set theory, graph theory, and algebraic structures.

To formulate problems precisely, solve the problems, apply formal proof techniques and explain the reasoning clearly

Course Outcomes:

1. To ***Acquire*** Knowledge of sets and logics for solving the real world problems
2. To ***Recognize*** and ***Analyze*** Relations, Functions, and their Characteristics
3. To ***apply*** advanced counting techniques such as the Inclusion-Exclusion principle and generating functions to solve complex counting problems and optimize problem-solving strategies.
4. To ***explore*** the concepts of group theory and their applications for solving the advance technological problems.
5. Illustrate the principles & concepts of graph theory for solving problems related to computer science.
6. Utilize tree structures as a modeling tool for solving algorithmic problems, demonstrating the ability to convert real-world problems into tree representations.

Guideline for Couse Activity:

The course coordinator should identify relative and innovative activities for course activity. Below are some suggested course activity for course coordinator

1. Mini Projects
2. Industry Visit
3. Seminar
4. Research Paper
5. Group Discussion

Course Contents

Mapping of Course Outcomes for Unit I	***CO1***

UNIT I	***Set TheoryandLogics***	***07Hours***

Introduction and Significance of Discrete Mathematics in Computer Engineering, Application areas in Computer Engineering. ***Set Theory***: Introduction to Set, Set Representation, Types of Sets, Power set, Set Operations, Principle of Inclusion and Exclusion. ***Logics and Proofs:*** Propositions, Conditional Propositions, Truth Tables, Tautology, Satisfiability, Contradiction, Algebra of proposition, Theory of Inference. Predicate Logic: First order predicate, well-formed formula of predicate, Universal and Existential Quantifiers, Translating English Statements into Propositions, Mathematical Induction.

Mapping of Course Outcomes for Unit II		*CO2*
UNIT II	***Relation & Function***	***07 Hours***

Relation: Definition of Relation, Properties of Binary Relations, Closure of Relations, Warshall‟s Algorithm,Equivalence Relations and Equivalence Classes, Partitions, Partial Ordering Relations (POSET), Hasse Diagrams and Lattices, Properties of lattices – Bounded, Complemented, Distributed, Modular and Complete lattice. ***Function***: Function Definition, Composition of Functions, Injective, Surjective and Bijective Function, Inverse of a Function, Growth of Functions.

Mapping of Course Outcomes for Unit III		*CO3*
UNIT III	***Counting Principles***	***07 Hours***

Introduction to Counting- rule of Sum and Product, Permutations and Combinations, Binomial Coefficients and Identities, Generalized Permutations and Combinations, Advanced Counting Techniques: Inclusion-Exclusion principle, Pigeonhole principle. Recurrence Relations: Basics of recurrence relations Solving linear recurrence relations.

Mapping of Course Outcomes for Unit IV		*CO4*
UNIT IV	***Group Theory***	***06Hours***

Definition, Basic Properties, Groups, Semi-group & Monoid, Abelian group, Subgroup, Normal subgroup, Groups and Coding, Group Homomorphism’s, Rings, Integral Domain and Field. ***Case Study:*** Application of Group Theory in Computer Engineering.

Mapping of Course Outcomes for Unit V		*CO5*
UNIT V	***Graph Theory***	***06 Hours***

Graph : Basic Terminology and Special Types of Graphs, Representation of graphs using adjacency matrix and adjacency list, , Paths and Circuits, Hamiltonian and Euler Paths and Circuits, Isomorphic Graphs, Planer Graph,Single source shortest path- Dijkstra's Algorithm, Graph coloring. ***Case Study:*** Applications of Graph Theory in Computer Engineering.

Mapping of Course Outcomes for Unit VI		*CO6*
UNIT VI	***Trees***	***06 Hours***

Introduction, properties of trees, Rooted Trees, Binary search tree, tree traversal, Prefix Codes, Huffman Algorithm for Optimal Tree, Spanning Trees, Minimum Spanning Trees, Kruskal‟s and Prim’s Algorithm for Minimum Spanning Tree. ***Case Study:*** Applications of Trees in Computer Engineering.

Text Books:

1. “ C. L. Liu, “Elements of Discrete Mathematics”‖, TMH, ISBN 10:0-07-066913-9.
2. “N. Biggs,“DiscreteMathematics”,3rdEd,OxfordUniversityPress,ISBN0–19-850717–8.
3. Kenneth H. Rosen, “Discrete Mathematics and its Applications”‖, Tata McGraw-Hill, ISBN 978- 0-07-288008-3
4. Narsingh Deo, “Graph with application to Engineering and Computer Science”, Prentice Hall of India, 1990, 0 – 87692 – 145 – 4

5. Eric Gossett, “Discrete Mathematical Structures with Proofs”, Wiley India Ltd, ISBN:978-81- 265-2758-8

E Books & Videos:

1. https://www.ebookphp.com/discrete-mathematical-structures-6th-edition-epub-pdf/
2. http://discrete.openmathbooks.org/pdfs/dmoi-tablet.pdf
3. http://home.iitk.ac.in/~arlal/book/mth202.pdf
4. https://web.stanford.edu/class/cs103x/cs103x-notes.pdf
5. http://home.iitk.ac.in/~arlal/book/mth202.pdf

MOOC/ Video Lectures available at:

1. https://www.nptel.ac.in/courses/106/106/106106094/
2. https://nptel.ac.in/courses/106/106/106106183/
3. https://nptel.ac.in/courses/106/103/106103205/
4. https://nptel.ac.in/courses/106/105/106105192/
5. https://nptel.ac.in/courses/111/106/111106050/
6. https://nptel.ac.in/courses/111/106/111106102/

CO25202: Object Oriented Paradigms		
TeachingScheme: TH:03 Hrs/Week PR 02 Hrs/Week	***Credit:03***	***Examination Scheme:*** Course Activity : 10Mark Mark In Semester:30Mark End Semester : 60 Mark Practical : 30 Mark

Prerequisite: Object oriented programming (OOP)

Course Objective:

To learn the basic concept of Java Programming.

To learn Object Oriented Programming(OOP) principles using Java Programming Language

To learn Exception handling concepts of Java Programming

To learn Applet and Multithreading concepts of Java Programming

To learn AWT concepts of Java Programming

Course Outcomes:

Students will be able to

1. Develop programs using Java , an Object Oriented Programming language.
2. Develop application using inheritance, encapsulation, and polymorphism.
3. Demonstrate Packages and Interfaces
4. Demonstrate Exception handling and file operations
5. Develop application using Multithreading for robust application development.
6. Develop application using Applet and AWT.

Course Activity :

The course coordinator should identify relative and innovative activities for course activity. Below are some suggested course activity for course coordinator

1. Mini Project using Java Language
2. Industry Visit
3. Seminar

CourseContents

Mapping of Course Outcomes for Unit I	***CO1***

UNIT I	***Introduction to Java***	***07 Hours***

Features of Java ,JDK Environment ,OOPs Concepts Class, Abstraction , Encapsulation, Inheritance, Polymorphism, Difference between C++ and JAVA, Structure of Java program ,Data types ,Variables ,Operators , Keywords , Decision Making (if, switch), Looping(for, while) ,Type Casting ,Array Creating an array Types of Array - One Dimensional arrays - Two Dimensional array ,String - Arrays , Methods.

Mapping of Course Outcomes for Unit II	***CO2***

UNIT II	***Classes, Object and Inheritance***	***07 Hours***

Creating Classes and objects , Memory allocation for objects , Constructor , Implementation of Inheritance Simple, Multilevel, Using super: Using super to Call Superclass Constructors, , Implementation of Polymorphism Method Overloading, Method Overriding ,Modifiers and Access

Control. Using final with Inheritance: Using final to Prevent Overriding , Using final to Prevent Inheritance

Mapping of Course Outcomes for Unit III		***CO3***
UNIT III	***Packages and Interfaces***	***07 Hours***
Packages : Defining a Package , Finding Packages and CLASSPATH, A Short Package Example Packages and Member Access : Example, Importing Packages, ***Interfaces :*** Defining an Interface, Implementing Interfaces, Nested Interfaces, Applying Interfaces, Variables in Interfaces, Interfaces Can Be Extended ***Default Interface Methods:*** Default Method Fundamentals, Example.		
Mapping of Course Outcomes for Unit IV		***CO4***
UNIT IV	***Exception and File Handling***	***06 Hours***
Exception: Exception types, Using try catch and multiple catch, Nested try, throw , throws and finally, Creating user defined Exceptions ***File Handling :***Stream, Byte Stream Classes, Character Stream Classes, File IO basics, File operations ,Creating file, Reading file (character, byte) ,Writing file (character, byte).		
Mapping of Course Outcomes for Unit V		***CO5***
UNIT V	***Multithreading in Java***	***06 Hours***
Concurrency and Synchronization, Java Thread Model: Thread priorities, Synchronization, Messaging, Main Thread, Creating thread: Implementing Thread using thread class and Runnable interface. Creating multiple threads using is Alive() and join().		
Mapping of Course Outcomes for Unit VI		***CO6***
UNIT VI	***Applet and AWT Programming***	***06 Hours***
Applet :Introduction, Types applet, Applet Life cycle, Creating applet, Applet tag, Applet Classes, Color, Graphics , Font. ***AWT :*** Components and container used in AWT, Layout managers, Listeners and Adapter classes, Event Delegation model		

Books and Other Resources

TextBooks:

1. Programming with JAVA - E Balgurusamy
2. Herbert Schildt, "The Complete Reference Java", 9th Ed, TMH,ISBN: 978-0-07-180856-9.

ReferenceBooks:

1. The Complete Reference – JAVA Herbert Schildt
2. " Maureen Spankle, "Problem Solving and Programming Concepts", Pearson; 9th edition, ISBN-10: 9780132492645, ISBN-13: 978- 0132492645

Guidelines for Term Work Assessment :

Term work assessment will be based on overall performance of Laboratory assignments performed by a students. Each Laboratory assignment assessment will assign grade/marks based on parameters, such as timely completion, performance, efficient codes, and punctuality.

Guidelines for Practical Examination :

Problem statements will be formed based on assignments and performance will be evaluated by Internal and External Examiner. During practical assessment, maximum weightage should be given to satisfactory implementation of the problem statement. Relevant questions may be asked at the time of evaluation to test the student‘s understanding of the fundamentals, effective and efficient implementation.

Guidelines for Laboratory Conduction :

A working computer system with either Windows or Linux (with minimum configurations- HDD:40 GB, RAM: 512M

Simple Editor Notepad or other editor tools like Edit plus or notepad++.

A web browser IE/ Firefox/ chrome. Apache Tomcat web server.

JVM(Java virtual machine) must be installed on your system

BDK(Bean development kit) must be also be installed

Practical Assignments

1 Write a java program to find the Fibonacci series using recursive and non-recursive functions

2 Write a java program that reads a line of integers and displays each integers and the sum of all integers use String Tokenizer

3 Write a java program to implement Interface using extends keyword.

4 Write a java program to create user defined package.

5 Department maintains a student information. Write a Java program to write records to the file and read students records from file demonstrating I/O management through File and stream classes.

6 A. Create a basic applet that functions as a simple calculator. The applet should accept two numbers and perform operations such as addition, subtraction, multiplication, and division based on user input.

7 Create a basic calculator using Java Swing that can perform basic arithmetic operations like addition, subtraction, multiplication, and division. The calculator should have buttons for numbers, operations, and a display area

8 Build a form for student registration that takes inputs such as name, age, address, and course. Add validation to ensure all fields are filled and the data is in the correct format.

9 Write a program demonstrating multithreading in Java : getName(), getPriority(),setPriority(), join(), isAlive(), synchronized method must be utilized.

10 Write a java program that connects to a database using JDBC and perform Insert, delete, update and display operations.

Mini Project:

Here are some key components mini-project should ideally contain:

Core Java Concepts and Connectivity

CO25203: DigitalElectronics and Computer Organization

Teaching Scheme:	***Credit: 04***	***Examination Scheme:***
TH: 03 Hrs/Week PR: 02 Hrs/Week		Course Activity: 10 Mark In-Semester: 20 Mark End-Semester: 60 Mark Oral : 30 Mark

Prerequisite:
Basis Electronics Engineering

Course Objective:

Understand number systems, Boolean algebra, and logic gates.
Design and analyze combinational logic circuits.
Implement sequential logic circuits using flip-flops, registers, and counters.
Understand computer organization and arithmetic operations.
Explain processor organization and control unit design.
Understand memory organization and input/output systems.

Course Outcomes:

1. Apply number system conversions and Boolean algebra to solve digital logic problems
2. Design and analyze combinational logic circuits.
3. Design and implement sequential logic circuits.
4. Explain computer organization and perform computer arithmetic operations.
5. Analyze processor organization, instruction execution, and control units.
6. Evaluate memory hierarchy and input/output organization.

Course Contents

Mapping of Course Outcomes for Unit I	***CO1***

UNIT I	***Number System, Logic Gate and Boolean Algebra***	***07 Hours***

Number systems and binary representation: Number system conversion, sign magnitude, 1's complement, 2's complement, subtraction using 1's and 2's complement, binary codes (Gray, BCD, Excess-3). Logic Gates: Truth tables and logic gates. ***Boolean Algebra:*** Basic theorems and laws, Boolean expression simplification, SOP, POS, and Karnaugh Map (K-Map)-2,3 and 4 Variable

Mapping of Course Outcomes for Unit II	***CO2***

UNIT II	***Combinational Circuits***	***06 Hours***

Introduction to combinational logic circuits and clock signals; design and implementation of Half Adder, Half Subtractor, Full Adder, Full Subtractor, BCD Adder, Parity Generator and Checker, Magnitude Comparator, Multiplexer, and Demultiplexer.

Mapping of Course Outcomes for Unit III	***CO3***

UNIT III	***Sequential Circuits***	***06 Hours***

Introduction to sequential logic circuits; flip-flops and their characteristics; types of flip-flops (SR, JK, D, and T); flip-flop conversions (JK to SR, JK to D, JK to T, and SR to JK); ***Registers*** and their types; shift registers; ***Counter:*** synchronous and asynchronous counters.

Mapping of Course Outcomes for Unit IV		*CO4*
UNIT IV	*Basic Structure of Computer and Computer Arithmetic*	*07 Hours*

Von Neumann Architecture, Functional units of Computer, Basic Operational Concept – Address, Data, Control Bus structure, Memory location and Address, Memory operations

Computer Arithmetic: Addition and Subtraction of Signed Numbers, Multiplication of Unsigned Numbers, Multiplication of signed Numbers, Booth Multiplication, Floating-Point Numbers and Operations.

Mapping of Course Outcomes for Unit V		*CO5*
07 Hours *UNIT V*	*Control Processing Unit*	

Processor: Organization, CPU Registers and Its types Instructions: Elements, Format and representation Addressing modes

Basic Processing Unit: Instruction execution cycle, hardware components of the basic processing unit, instruction pipelining and pipeline hazards, control signals, hardwired and microprogrammed control units, and single bus organization.

Mapping of Course Outcomes for Unit VI		*CO6*
UNIT VI	*Memory and I/O System*	*07 Hours*

Memory Systems: Characteristics of Memory Systems, Memory Hierarchy, and Memory read & write cycle, Semiconductor RAM Memories, Read-only Memories. Cache Memory – Principle of Locality, Organization, Mapping functions, write policies, Cache Coherence. Input / Output Systems: Accessing I/O devices, I/O Module-Programmed I/O, Interrupt Driven I/O, Direct Memory Access (DMA).

Books and Other Resources

Text Books:

1. "Modern Digital Electronics", R.P. Jain, Tata McGraw-Hill, Third Edition
2. "Computer organization and architecture, designing for performance" by William Stallings , Prentice Hall ,Eighth edition

Reference Books:

1. "Digital Design", M Morris Mano, Prentice Hall, Third Edition
2. "Computer organization" , Hamacher and Zaky, Fifth Edition

Practical's

Guidelines for Students Journal

The laboratory assignments are to be submitted by student in the form of journal. Journal consists of prologue, Certificate, table of contents, and handwritten write-up of each assignment (Title, Objectives, Problem Statement, Outcomes, software & Hardware requirements, Date of Completion, Assessment grade/marks and assessor's sign, Theory Concept, circuit diagram, pin configuration, conclusion/analysis). ***As a conscious effort and little contribution towards Green IT and environment awareness, attaching printed papers as part of write-ups and***

program listing to journal may be avoided.

Guidelines for Laboratory /TW Assessment

Continuous assessment of laboratory work is done based on overall performance and Laboratory performance of student.

Each Laboratory assignment assessment should assign grade/marks based on parameters with appropriate weightage.

Suggested parameters for overall assessment as well as each Laboratory assignment assessment include- timely completion, performance, innovation, efficiency, punctuality and neatness.

Guidelines for Laboratory Conduction

The instructor is expected to frame the assignments by understanding the prerequisites, technological aspects, utility and recent trends related to the topic. The assignment framing policy need to address the average students and inclusive of an element to attract and promote the intelligent students. The instructor may set multiple sets of assignments and distribute among batches of students.

Practical Assignments

1. Design and Implement Full Adder using IC74138
2. Design and implement a 4-bit Binary-to-Gray Code Converter (IC7486).
3. Design and Realization of BCD Adder using 4-bit Binary Adder (IC 7483).
4. Design and implement Even/Odd Parity Generator and Parity Checker using EX-OR gates (IC7486)
5. Design and implement Flip-Flop conversions: JK to SR/JK to D/JK to T/SR to JK (IC7474/IC7476)
6. Design and implement a 4-bit Synchronous Up/Down Counter. (74LS193 / 74LS192/Flip flop)
7. Develop a C++/Python/Java program to implement Booth's Multiplication Algorithm.
8. Study and analyze the organization of a Single Bus Processing Unit and Microinstruction Execution (Case Study/Simulation).

HS25201: Public Speaking and Aptitude

Teaching Scheme:	Credits: 2	Examination Scheme:
Th.: 1 Hr. Pr.: 2 Hrs.		Activity: 10 Marks TW: 30 Marks OR: 30 Marks

Course Objectives:

1. To develop effective public speaking styles through conversational and communication skills and also enhance English speaking skills by focusing on body language and understanding the situational requirements for effective public speaking
2. To develop students' quantitative, logical and analytical abilities required to solve aptitude-based problems commonly encountered in competitive examinations and also enhance their problem-solving speed, decision-making ability and logical reasoning skills

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

1. Communicate effectively in various public speaking situations and deliver organised and engaging speeches with appropriate body language, voice modulation and confident speech techniques

2. Apply appropriate quantitative, logical, and reasoning strategies to efficiently solve numerical aptitude, data interpretation, and logical reasoning problems with improved speed and accuracy in placement and competitive examination contexts and apply rapid analytical, logical and decision-making strategies to solve time-bound problems with improved accuracy and efficiency

Course Contents:

Unit 1: Spoken English **(7 Hrs.)**

Pre-Assessment, Vocabulary made easy, the Power of Words, Introduction to Word Accent, Introduction to Rhythm: Intonation, Rising Intonation, Falling Intonation, Role Playing: Specific scenarios, Telephone Skills: Making and Receiving Calls, Voice, Intonation and Language, Conversations: The Role and Types of Questions

Unit 2: Impactful Presentations **(6 Hrs.)**

Body Language: Introduction, Mechanics & Style, Voice Modulation: Voice Projection, Avoiding Fillers and Emphasis, Power of Pause: Pause to engage audience in Conversation, Combine Pause & Repetition Techniques, Demonstrate Confidence, Control and establish Presence, Empathy: Practice Heartfelt Communication, Impact of Communication and how to deliver impactful speech

Unit 3: General Aptitude for all Competitive Exams **(13 Hrs.)**

Quantitative Aptitude: Data interpretation: data graphs (bar graphs, pie charts and other graphs representing data), 2- and 3-dimensional plots, maps and tables, Numerical computation and estimation: ratios, percentages, powers, exponents and logarithms, permutations and combinations, series Mensuration and geometry Elementary statistics and probability

Analytical Aptitude: Logic: deduction and induction, Analogy, Numerical relations and reasoning

Spatial Aptitude: Transformation of shapes: translation, rotation, scaling, mirroring, assembling and grouping paper folding, cutting and patterns in 2 and 3 dimensions

Textbooks

1. *A Course in Phonetics and Spoken English* by T. Balasubramanian
2. *Effective Technical Communication* by M. Ashraf Rizvi
3. *Quantitative Aptitude for Competitive Examinations* by R.S. Aggarwal

VPKBIET BARAMATI Affiliated to S.P.P.U. Pune Id No./PU/PN/Engg./152/2000 PUNE (MS.)

Reference Books

1. *High School English Grammar and Composition* by Wren & Martin

2. *How to Speak, How to Listen* by Mortimer J. Adler

3. *Logical and Analytical Reasoning* by A.K. Gupta

NPTEL Course: Mastering Speaking and Presentations: A case Based Approach by Prof. Seema Singh, IIT, Kharagpur https://onlinecourses.nptel.ac.in/noc25_hs96/preview

Lab Session Activities Plan

Session 1: Vocabulary & Word Accent

o Pre-Assessment: Conduct a quick spoken test to evaluate pronunciation and fluency.

o Vocabulary exercises using flashcards or interactive word-building games

Session 2: Intonation & Rhythm in Speech

o Introduction to rising and falling intonation with examples

o Role-play exercises to practice intonation in different scenarios: expressing surprise, asking questions, etc.

Session 3: Telephone Skills & Professional Conversations

o Practicing making and taking calls with simulated dialogues

o Focus on voice modulation, clarity and polite expressions

Session 4: Body Language & Stage Presence

o Practicing posture, movement and eye contact while speaking.

o Group feedback and analysis in classroom

Session 5: Empathy & Heartfelt Communication

o Interactive storytelling to practice emotional connection

o Speech practice: delivering a short talk with an emotional appeal

Session 6: Quantitative Aptitude –Data Interpretation & Computation

o Solving numerical problems based on bar graphs, pie charts and tables

o Quick estimation exercises using ratios, percentages and logarithms

o Group challenges on permutations and combinations

Session 7: Analytical Aptitude – Logical & Numerical Reasoning

o Deduction and induction puzzles

o Solving analogy-based reasoning questions

o Speed tests for numerical relations and reasoning

Session 8: Spatial Aptitude – Shape & Pattern Recognition

o Hands-on paper folding and cutting exercises

o Visualization tasks for rotation, scaling, and mirroring of shapes

o Solving pattern-based problems in 2D and 3D space

Course Coordinator **Dean Academics** **Principal**

VPKBIET BARAMATI Affiliated to S.P.P.U. Pune Id No./PU/PN/Engg./152/2000 PUNE (MS.)

CO25204 : Data Analytics and Visualization using Python

Teaching Scheme:	***Credit: 02***	***ExaminationScheme:***
TH: 01 Hrs/Week	TH Credit :01	CourseActivity:40 Mark
PR: 02 Hrs/Week	PR Credit :01	Practical: 30 Mark

Prerequisite: Students are expected to have a good understanding of Programming and Problem Solving(PPS) Python Programming.

Course Objective:

To understand principles of Data Analytics and Visualization for the analysis of real time problems

To develop in depth understanding and implementation of the key technologies in Data Science and data analytics to understand the Processor and Instructions

To develop ability to analyze and demonstrate knowledge of statistical data analysis techniques for decision-making

To gain practical, hands-on experience with statistics programming languages.

Course Outcomes:

1. Understand and apply basic Python programming concepts for data handling and analysis
2. Perform exploratory data analysis (EDA) and visualize data using Matplotlib and Seaborn.
3. Interpret data insights from visualizations and summarize data effectively.
4. Apply principles of Data Science for the analysis of real time problems

Course Activity :

The course coordinator should identify relative and innovative activities for course activity. Below are some suggested course activity for course coordinator

1. Data Analytics Mini Project
2. Collection of Dataset on different domains and analyze that data.
3. Industry Visit
4. Seminar
5. Research Paper in data analytics and visualization domain

Course Contents

Mapping of Course Outcomes for Unit I		***CO1***
UNIT I	***Introduction to Data Analysis and Basics of Python***	***04 Hours***

Introduction to Data Analytics : Data Analytics, Importance and Real-life Applications, Types of Data: Structured vs. Unstructured, ***Python Basics for Data Handling*** :Data Types (Numbers, Strings, Lists, Tuples, Dictionaries),Conditional Statements and Loops, Functions and Modules in Python, ***Data Handling with Pandas*** :Introduction to Pandas, Creating and Manipulating DataFrames, Reading and Writing CSV Files.

Mapping of Course Outcomes for Unit II		***CO2***
UNIT II	***Exploratory Data Analysis (EDA) & Basic Visualization***	***04 Hours***

Understanding Data : Data Cleaning (Handling Missing Values), Removing Duplicates, Basic Summary Statistics (Mean, Median, Mode), Outliers, Data Normalization , Data Transformation , ***Basic Data Visualization with Matplotlib*** :Line Plot, Bar Chart, Histogram, ***Basic Data Visualization with Seaborn*** : Scatter Plot, Box Plot, Pair Plot.

Mapping of Course Outcomes for Unit III	*CO3*	
UNIT III	*Data Interpretation & Simple Plotting*	*04 Hours*

Interpreting Data from Plots :Identifying Trends and Patterns, Understanding Data Distributions, Comparing Data using Graphs, ***Plot Customization in Matplotlib*** :Adding Titles and Labels, Adjusting Colors and Styles, Displaying Multiple Plots, ***Simple Data Aggregation***: Grouping Data with Pandas, Calculating Averages and Totals, Sorting and Filtering Data.

Books and Other Resources

Text Books:

"Python for Data Analysis" – ***Wes McKinney***

"Matplotlib and Seaborn for Data Visualization" – ***Allen Yu, Claire Chung***

Reference Books:

"Data Science from Scratch" – ***Joel Grus***

"Python Data Science Handbook" – ***Jake VanderPlas***

"Think Stats: Exploratory Data Analysis in Python" – ***Allen B. Downey***

Chirag Shah, "A Hands-On Introduction To Data Science", Cambridge University Press,(2020), ISBN : ISBN 978-1-108-47244-9. 2. Wes McKinney,

"Python for Data Analysis", O' Reilly media, ISBN : 978-1-449-31979-3. 3. "Scikit-learn Cookbook", Trent hauk,Packt Publishing, ISBN: 9781787286382 4. R Kent Dybvig,

"The Scheme Programming Language", MIT Press, ISBN 978-0-262-51298-5. 5. Jenny Kim, Benjamin Bengfort, "Data Analytics with Hadoop", OReilly Media, Inc.

Guidelines for Term Work Assessment :

Term work assessment will be based on overall performance of Laboratory assignments performed by a students. Each Laboratory assignment assessment will assign grade/marks based on parameters, such as timely completion, performance, efficient codes, and punctuality.

Guidelines for Practical Examination :

Problem statements will be formed based on assignments and performance will be evaluated by Internal and External Examiner. During practical assessment, maximum weightage should be given to satisfactory implementation of the problem statement. Relevant questions may be asked at the time of evaluation to test the student's understanding of the fundamentals, effective and efficient implementation..

Guidelines for Laboratory Conduction :

Use of open source software is encouraged. Based on the concepts learned.

Operating System recommended: - 64-bit Open source Linux or its derivative Programming, Windows.

Tools recommended: - Latest version of Python, Anaconda (Jupyter Notebook), Visual Studio, PyCharm.

Practical Assignments

1. Data Wrangling I

   Perform the following operations using Python on any open source dataset (e.g., data.csv)

   - Import all the required Python Libraries.
   - Locate an open source data from the web (e.g., https://www.kaggle.com). Provide a clear description of the data and its source (i.e., URL of the web site).
   - Load the Dataset into pandas dataframe.
   - Data Preprocessing: check for missing values in the data using pandas isnull(), describe() function to get some initial statistics. Provide variable descriptions. Types of variables

etc.Check the dimensions of the data frame.

- Data Normalization: Summarize the types of variables by checking the data types (i.e., character, numeric, integer, factor, and logical) of the variables in the data set. If variables are not in the correct data type, apply proper type conversions.
- Turn categorical variables into quantitative variables in Python.

2. Data Wrangling II

Create an "Academic performance" dataset of students and perform the following operations using Python.

- Scan all variables for missing values and inconsistencies. If there are missing values and/or inconsistencies, use any of the suitable techniques to deal with them.
- Scan all numeric variables for outliers. If there are outliers, use any of the suitable techniques to deal with them.

3. Data Wrangling III

Perform the following statistic operations on any open source dataset (e.g., data.csv)

Provide summary statistics (mean, median, minimum, maximum, standard deviation) for a dataset (age, income etc.) with numeric variables grouped by one of the qualitative (categorical) variable. For example, if your categorical variable is age groups and quantitative variable is income, then provide summary statistics of income grouped by the age groups. Create a list that contains a numeric value for each response to the categorical variable.

Write a Python program to display some basic statistical details like percentile, mean, standard deviation etc. of the species of 'Iris-setosa', 'Iris-versicolor' and 'Iris-versicolor'of iris.csv dataset. Provide the codes with outputs and explain everything that you do in this step.

Usedataset https://www.kaggle.com/c/boston-housing for 4-6 assignments

4. Write a Python program to demonstrate how to draw a Bar Plot using Matplotlib/Seaborn and analyze the data patterns if any.

5. Write a Python program to demonstrate how to draw a Scatter Plot using Matplotlib/Seaboarn and analyze the data patterns if any.

6. Write a Python program to demonstrate how to draw a Histogram Plot using Matplotlib/Seaboarn and analyze the data patterns if any.

7. Write a Python program to demonstrate how to draw a Pie Chart using Matplotlib/Seaboarn Using iris.csv dataset and analyze the data patterns if any.

8. Use the dataset 'titanic',

1. Plot a box plot for distribution of age with respect to each gender along with the information about whether they survived or not. (Column names : 'sex' and 'age')
2. Write observations on the inference from the above statistics.

Semester II

GS25201: Advanced Mathematics for Computer Engineering		
Teaching Scheme: Theory : 3 Hours/Week	***Credit: 03***	***Examination Scheme:*** Course Activity: 10 Mark In-Semester : 30 Marks End-Semester : 60 Marks

Prerequisite: Differential & Integral calculus, Taylor series, Differential equations of first order and first degree, Fourier series, Collection, Classification & Representation of data.

Course Objective:

To provide the students with concepts and techniques in Linear differential equations, Fourier transform, Statistical methods, and Probability theory. The aim is to equip them with the techniques to understand advanced-level mathematics and its applications that would be useful in their discipline and enhance their thinking power.

Course Outcomes:

1. Solve higher-order linear differential equations using appropriate techniques useful for modeling in their field.
2. Understand the concepts of Fourier transform
3. Understand and apply the various concepts of statistical methods of correlation, and regression and Apply them in their field.
4. Apply the concepts of appropriate Probability and Probability distribution for data analysis and predictions in multiple data sets.
5. Solve Algebraic, Transcendental equations and System of linear equations using numerical techniques.
6. Compute Interpolating polynomials, numerical differentiation, and integration, numerical solutions of ordinary differentialequationsusedin modernscientificcomputing.

Course Contents

Mapping of Course Outcomes for Unit I		***CO1***
UNIT I	***Linear Differential Equations (LDE) and Applications***	***07 Hours***

Introduction, Solution of LDE, General method, short-cut method, Method of variation of parameters, Cauchy's, Legendre's DE, Simultaneous DE.

Mapping of Course Outcomes for Unit II		***CO2***
UNIT II	***Fourier Transform and Statistics***	***07 Hours***

Fourier Transform: General Fourier, Fourier Sine, Cosine, and inverse transforms.
Statistics: Measures of dispersion, Moments, Skewness and Kurtosis, Correlation and Regression analysis.

Mapping of Course Outcomes for Unit III		***CO3***
UNIT III	***Regression Models***	***07 Hours***

Importance of Regression in Data Mining, Simple Linear Regression, Model: $Y = \beta0 + \beta1X + \varepsilon$ Assumptions, Estimation of $\beta0$ and $\beta1$ by the method of least squares, Multiple linear regression model $Y = \beta0 + \beta1X1 + ... + \beta pXp + \varepsilon$, residuals, Least-Squares Estimation of the Regression Coefficients, obtaining normal equations, Solutions of normal equations, Generalized linear models, and applications.

Mapping of Course Outcomes for Unit IV		*CO4*
UNIT IV	***Probability and Probability Distributions***	***07Hours***

Theorems on probability, Random variables, Probability Mass function, Probability Density function, Mathematical Expectation. Binomial, Poisson, and Normal distribution and applications.

Mapping of Course Outcomes for Unit V		*CO5*
UNIT V	***Numerical Methods***	***07 Hours***

Numerical Solution of Algebraic and Transcendental Equations: Bisection, Secant, Regula-Falsi, Newton–Raphson and Successive Approximation Methods.
Numerical Solutions of System of linear equations: Gauss elimination, LU Decomposition, Cholesky, Jacobi and Gauss-Seidel Methods

Mapping of Course Outcomes for Unit VI		*CO6*
UNIT VI	***Numerical Methods***	***07 Hours***

Interpolation: Finite Differences, Newton's and Lagrange's Interpolation formula, Numerical Differentiation.
Numerical Integration: Trapezoidal and Simpson's rules.
Solution of Ordinary differential equations: Euler's, Modified Euler's, Runge-Kutta 4th order.

Books and Other Resources

Text Books:

1. Higher Engineering Mathematics by B.V. Ramana (Tata McGraw-Hill).
2. Higher Engineering Mathematics by B. S. Grewal (Khanna Publication, Delhi).

E Books & Videos:

1. Erwin Kreyszig, "Advanced Engineering Mathematics", 10ed, Wiley India
2. M. D. Greenberg, "Advanced Engineering Mathematics", 2nd e Pearson Education
3. Peter V. O'Neil, "Advanced Engineering Mathematics", 7ed, Cengage Learning
4. S. L. Ross, "Differential Equations", 3e, Wiley India
5. Sheldon M. Ross, "Introduction to Probability and Statistics for Engineers and Scientists", 5e, Elsevier Academic Press
6. M. K. Jain, S. R. K. Iyengar, and R. K. Jain, "Numerical Methods for Scientific and Engineering Computation", 5e, (New Age International Publication).
7. Draper, N. R. and Smith, H. "Applied Regression analysis", (1998) (John Wiley) Third Edition.
8. S.P. Gupta, Sultan Chand and Sons, "Statistical Methods", New Delhi, 2009.

Guidelines for Term Work Assessment :

1. Tutorials for the subject shall be engaged in a minimum of three batches (batch size of 22 students) per division.
2. Term work shall consist of six assignments on each unit and is based on performance and continuous internal assessment.

CO25211 : Data Structures and Algorithms

Teaching Scheme:	***Credit: 04***	***Examination Scheme:***
TH: 03 Hrs/Week	TH Credit :03	CourseActivity: 10 Mark
PR: 02 Hrs/Week	PR Credit :01	InSemester: 30 Mark
		End Semester: 60 Mark
		Practical: 30 Mark

Prerequisite: Students are expected to have a good understanding of Discrete Mathematics, Programming and Problem Solving

Course Objective:

To develop a logic for graphical modeling of the real life problems.

To suggest appropriate data structure and algorithm for graphical solutions of the problems.

To understand advanced data structures to solve complex problems in various domains.

To operate on the various structured data.

To build the logic to use appropriate data structure in logical and computational solutions.

Course Outcomes:

1. Design and apply the objectives and advantages of an effective hashing scheme for practical applications
2. Utilize non-linear data structures to address problems across different domains
3. Apply non-linear data structures for solving real time problems in various domains.
4. Design and apply advanced data structures in the application development
5. Develop and apply algorithms for insertion, deletion, and search operations in B-Trees and B+ Trees
6. Apply appropriate data structures and algorithms for efficient file storage and retrieval

Course Activity :

The course coordinator should identify relative and innovative activities for course activity. Below are some suggested course activity for course coordinator

1. Mini Project using Data Structures
2. Seminar
3. Research Paper in Data Structures domain
4. NPTEL course on Data Structures

Course Contents

Mapping of Course Outcomes for Unit I		***CO1***
UNIT I	***Hashing***	***07 Hours***

Hash Table- Concepts-hash table, hash function, basic operations, bucket, collision, probe, synonym, overflow, open hashing, closed hashing, perfect hash function, Collision resolution strategies- open addressing and chaining, Hash table overflow- open addressing and chaining, extendible hashing, closed addressing and separate chaining.

Skip List- representation, searching and operations- insertion, removal

Mapping of Course Outcomes for Unit II		***CO2***
UNIT II	***Trees***	***07 Hours***

Tree- basic terminology, General tree and its representation, representation using sequential and linked organization, Binary tree- properties, converting tree to binary tree, binary tree traversals(recursive and non-recursive)- inorder,

preorder, post order, depth first and breadth first, Operations on binary tree. ,inary Search Tree (BST), BST operations, Threaded binary search tree- concepts, threading, insertion and deletion of nodes in threaded binary search tree

Mapping of Course Outcomes for Unit III	***CO3***

UNIT III	***Graphs***	***07 Hours***

Basic Concepts, Storage representation, Adjacency matrix, adjacency list, adjacency multi list, inverse adjacency list. Traversals-depth first and breadth first, Minimum spanning Tree, Greedy algorithms for computing minimum spanning tree- Prims and Kruskal Algorithms, Dikjtra's Single source shortest path.

Mapping of Course Outcomes for Unit IV	***CO4***

UNIT IV	***Search Trees***	***07Hours***

Symbol Table-Representation of Symbol Tables- Static tree table and Dynamic tree table, Weight balanced tree - Optimal Binary Search Tree (OBST), OBST as an example of Dynamic Programming, Height Balanced Tree- AVL tree. Red-Black Tree, AA tree, K-dimensional tree, Splay Tree

Mapping of Course Outcomes for Unit V	***CO5***

UNIT V	***Indexing and Multiway Trees***	***07 Hours***

Indexing and Multiway Trees- Indexing, indexing techniques-primary, secondary, dense, sparse, Multiway search trees, B-Tree- insertion, deletion, B+Tree - insertion, deletion, use of B+ tree in Indexing, Trie Tree.

Mapping of Course Outcomes for Unit VI	***CO6***

UNIT VI	***File Organization***	***07 Hours***

Files: concept, need, primitive operations. Sequential file organization- concept and primitive operations, Direct Access File- Concepts and Primitive operations, Indexed sequential file organization-concept, types of indices, structure of index sequential file, Linked Organization- multilist files, coral rings, inverted files and cellular partitions.

Books and Other Resources

Text Books:

1. Horowitz and Sahani, “Fundamentals of Data Structures in C++”, University Press, ISBN 10: 0716782928 ISBN 13: 9780716782926.
2. Michael T. Goodrich, Roberto Tamassia, Michael H. Goldwasser, “Data Structures and Algorithms in Python”, Wiley Publication, ISBN: 978-1-118-29027-9
3. M Folk, B Zoellick, G. Riccardi, “File Structures‖, Pearson Education”, ISBN:81-7758-37-5
4. Peter Brass, “Advanced Data Structures”‖, Cambridge University Press, ISBN: 978-1-107-43982-5

Reference Books:

1. A. Aho, J. Hopcroft, J. Ulman, “Data Structures and Algorithms”‖, Pearson Education, 1998, ISBN-0-201-43578-0.
2. Michael J Folk, “File Structures an Object Oriented Approach with C++‖”, Pearson Education, ISBN: 81-7758-373-5.
3. Brassard and Bratley, “Fundamentals of Algorithmic”, Prentice Hall India/Pearson Education, ISBN
4. Sartaj Sahani, “Data Structures, Algorithms and Applications in C++”‖, Second Edition, University Press, ISBN:81-7371522 X.
5. G A V Pai, “Data Structures and Algorithms”‖, McGraw-Hill Companies, ISBN -9780070667266.

Guidelines for Term Work Assessment :

Term work assessment will be based on overall performance of Laboratory assignments performed by a students. Each Laboratory assignment assessment will assign grade/marks based on parameters, such as timely completion, performance, efficient codes, and punctuality.

Guidelines for Practical Examination :

Problem statements will be formed based on assignments and performance will be evaluated by Internal and External Examiner. During practical assessment, maximum weightage should be given to satisfactory implementation of the problem statement. Relevant questions may be asked at the time of evaluation to test the student‘s understanding of the fundamentals, effective and efficient implementation..

Guidelines for Laboratory Conduction :

Use of open source software is encouraged. Based on the concepts learned.
Operating System recommended :- 64-bit Open source Linux or its derivative Programming

Practical Assignments

1. Implement binary tree and perform tree traversals non-recursively.

2. Implement binary search tree and perform following operations on it:
   a. create
   b. display
   c. insert. height
   d. mirror image
   e. delete
   f. leaf nodes

3. Convert a given binary tree into threaded binary tree and apply traversals on it.

4. Represent a given graph using adjacency matrix/ adjacency list to perform DFS and BFS.

5. Implementation of prims/kruskal’s algorithm to find minimum spanning tree of a given graph.

6. Implementation of dijkstra’s algorithm to find shortest path between two nodes.

7. Implementation of heap sort.

8. Implementation of hash table and handle collisions using linear probing.

9. Department maintains a student information. The file contains roll number, name, division and address. Allow user to add, delete information of student. Display information of particular employee. If record of student does not exist an appropriate message is displayed. If it is, then the system displays the student details. Use sequential file to main the data.

10. Design a mini project using data structure and file handling concepts

CO25212TH: Microprocessor

Teaching Scheme:	***Credit:04***	***Examination Scheme:***
TH: 03 Hrs/Week	TH Credit :03	Course Activity: 10 Mark
PR: 02 Hrs/Week	PR Credit : 01	In-Semester: 30 Mark
		End-Semester: 60 Mark
		Oral: 30 Mark

Prerequisite:
Digital Electronics and Logic Design

Course Objectives:

To understand architecture and programming of 8-bit, 16-bit and 32-bit processors.
To develop assembly language programs for 8085 and 8086.
To interface memory and peripherals using standard chips.
To study 8051 microcontroller for embedded applications.
To understand advanced architecture concepts of 80386.

Course Outcomes:

CO1: Explain architecture and internal organization of 8085, 8086, 8051 and 80386.
CO2: Write assembly language programs for arithmetic and logic operations.
CO3: Interface memory and I/O devices.
CO4: Analyze interrupt handling and timing diagrams.
CO5: Understand protected mode and memory management concepts.
CO6: Exhibit ability to apply 8051 instruction set and addressing modes to develop simple assembly language programs.

Guideline for Couse Activity:
The course coordinator should identify relative and innovative activities for course activity. Below are some suggested course activity for course coordinator

1. Mini Project
2. Survey on uses of Microprocessor with emerging technology
3. Industry Visit
4. Poster/Technical Seminar
5. Research Paper
6. Intellectual Property Rights

Mapping of Course Outcomes for Unit I	***CO1***

UNIT I	***Introduction to 8085 Microprocessor***	***07 Hours***

Basics of microprocessor and applications, architecture and Functional Block Diagram, pin diagram of 8085, Bus Organization (Address, Data, Control Bus), timing concepts, instruction set and addressing modes, Instruction Cycle & Timing Diagram, Machine Cycles, stack and interrupts, overview of memory and I/O interfacing. Interfacing with Intel 8255.

Mapping of Course Outcomes for Unit II		*CO1, CO2*
UNIT II	***Introduction to 8086 and Assembly Language Programming***	***07 Hours***
Need and evolution of microprocessors with overview of 8086, Comparison: 8085 vs 8086, architecture and addressing modes, instruction set classification, basics of assembly language programming including syntax, arithmetic and loop instructions, procedures, macros, and string and array processing. Introduction To Pipelining, Interrupts of 8086		
Mapping of Course Outcomes for Unit III		*CO3*
UNIT III	***80386 System Architecture and Organization***	***07 Hours***
Comparison: 8086 vs 80386,80386 architecture, 32-bit Register Organization, Real Mode vs Protected Mode, Memory Management, functional pin diagram, memory and I/O organization, basic read/write cycles, processor state after reset.		
Mapping of Course Outcomes for Unit IV		*CO4*
UNIT IV	***Memory Management***	***06 Hours***
Global Descriptor Table, Local Descriptor Table, Interrupt Descriptor Table, GDTR, LDTR, IDTR. Formats of Descriptors and Selector, Segment Translation, Page Translation, Combining Segment and Page Translation.		
Mapping of Course Outcomes for Unit V		*CO5*
UNIT V	***Introduction to Microcontrollers***	***06 Hours***
8051 Microcontroller: Microprocessor Vs Microcontroller, Embedded Systems, Embedded Microcontrollers, Classification, 8051 Architecture- Registers, Pin diagram, I/O ports functions, Internal Memory organization. External Memory (ROM & RAM) interfacing, Timers & Counters, Serial Communication		
Mapping of Course Outcomes for Unit VI		*CO6*
UNIT VI	***8051 Instruction Set and Assembly Language Programming***	***06 Hours***
8051 Instruction Set: Addressing Modes, Data Transfer instructions, Arithmetic instructions, Logical instructions, Branch instructions, Bit manipulation instructions. Simple Assembly language program examples (without loops) to use these instructions. Interfacing Applications (LED, Seven Segment, Stepper Motor)		

Books and Other Resources

Textbooks:

1. A.Ray, K.Bhurchandi, "Advanced Microprocessors and peripherals: Arch, Programming & Interfacing", Tata McGraw Hill,2004 ISBN 0-07-463841-6
2. Douglas Hall, "Microprocessors & Interfacing", McGraw Hill, Revised 2 Edition, 2006 ISBN 0- 07-100462-9
3. Ramesh S. Gaonkar – ***Microprocessor Architecture, Programming and Applications with 8085***

4. Mazidi & Mazidi – ***The 8051 Microcontroller and Embedded Systems***

Reference Books:

1. Walter A. Triebel, “The 80386Dx Microprocessor: Hardware”, Software, and Interfacing, Pearson Education, ISBN: 0137877307, 9780137877300.
2. Brey, Barry B, “8086/8088, 80286, 80386 and 80486 Assembly Language Programming”,
   a. Prentice Hall, ISBN: 13: 9780023142475.
3. Mohammad Rafiquzzaman, "Microprocessors: Theory and Applications: Intel and Motorola", Prentice Hall, ISBN:-10:0966498011, 13:978:0966498011.
4. Introduction to 64 bit Intel Assembly Language Programming for Linux, 2nd Edition, Ray Seyfarth, ISBN10: 1478119209, ISBN-13: 9781478119203, 2012.Assembly Language Step-by-step: Programming with Linux, 3rd Edition, Jeff Duntemann, Wiley ISBN:-10 0470497025, ISBN-13: 978-0470497029, 2009.

Virtual Laboratory:

1. http://209.211.220.205/vlabiitece/mi/MI3.php

MOOC/ Video Lectures available at:

1. https://nptel.ac.in/courses/106/108/106108100/
2. https://nptel.ac.in/courses/108/107/108107029/

Guidelines for Term Work Assessment :

Term work assessment will be based on overall performance of Laboratory assignments performed by a students.

Guidelines for Practical Examination :

Problem statements will be formed based on assignments and performance will be evaluated by Internal and External Examiner. Relevant questions may be asked at the time of evaluation to test the student‘s understanding of the fundamentals, effective and efficient implementation

CO25212PR : Practical Assignments

1. Study of 8085 Trainer Kit
2. Write an X86/64 ALP to accept a string and to display its length.
3. Write an X86/64 ALP to find the largest of given Byte/Word/Dword/64-bit numbers
4. Write an X86/64 ALP to count the number of positive and negative numbers from the array.
5. Write an X86/64 ALP to multiply two 8 and 16 bit signed and unsigned numbers.
6. Write an X86/64 ALP to compare two strings using string instructions.
7. Write x86 ALP a) Ad d i t io n b) S u bt r a c t i o n c) M u l t ip l i c a t io n d) D i v i s io n
8. Write a 8051 ALP program for LED Blinking
9. Write a 8051 ALP program for LCD interfacing
10. Write a 8051 ALP program for Stepper Motor Interfacing

CO25213: Operating Systems

Teaching Scheme:	***Credit: 04***	***Examination Scheme:***
TH: 03 Hrs/Week PR: 02 Hrs/Week	TH Credit :03 PR Credit :01	Course Activity: 10 Mark In-Semester: 30 Mark End-Semester: 60 Mark Oral: 30 Mark

Prerequisite: Digital Electronics and Computer Organization, Programming Languages

Course Objective:

To learn the basic concepts operating system
To learn processes life cycle, process states and scheduling algorithms
To learn Inter Process synchronization, and communication between processes
To explore memory management, and virtual memory management policies..
To understand the organization and management of file systems,
To understand device management

Course Outcomes:

1. To Build the basic knowledge of Operating System
2. To identify process management strategies and processor scheduling algorithms
3. To apply concepts of Inter Process synchronization, and communication
4. To apply memory management, and virtual memory management strategies
5. To make use of concepts of File management and free space management
6. To make use of concepts of device management

Course Activity:

Course teacher will plan the course activity

Course Contents

Mapping of Course Outcomes for Unit I	***CO1***

UNIT I	***Operating System Structures***	***07 Hours***

Introduction to Operating System: Evolution of OS, Functions of OS, Operating System Components, O.S. Services, Types of OS, Kernel and types of Kernel, System Calls, Virtual Machines, Boot Sequence. Case study of UNIX operating System

Mapping of Course Outcomes for Unit II	***CO2***

UNIT II	***Processes Management***	***07 Hours***

Process concept: Creation, Termination of process, Process states, Context Switching, Process Control Block, Thread***:*** Concept of a Thread, Thread libraries, Multithreading, Comparison of thread
Process Scheduling, Scheduling criterion, Scheduling algorithms: FCFS, SJF, RR, Priority, Process System calls. Case Study of Unix Process Management

Mapping of Course Outcomes for Unit III	***CO3***

UNIT III	Inter process Communication & Synchronization	07 Hours

Synchronization: Critical section problem, Hardware support for mutual exclusion, Semaphores, Monitors, Classical Problems in Synchronization: Producer-consumer, Reader-writer,
Deadlock: Deadlock-principle, Deadlock prevention Deadlock avoidance, Deadlock detection and recovery, Case Study of Unix IPC

Mapping of Course Outcomes for Unit IV	CO4

UNIT IV	Memory management	06 Hours

Memory Management: Continuous and Non Contiguous memory management, Swapping, Paging, Segmentation. ***Virtual Memory Management***: Demand Paging, Page replacement algorithms- FIFO, LRU, Optimal, Thrashing, Allocation method, Case Study of Unix Memory management

Mapping of Course Outcomes for Unit V	CO5

UNIT V	File Management	06 Hours

File Organization: Concept of files, File Attributes, File operations, File types, Directories and types of directories, Free space management. ***File System Implementation:*** Data structures like Inode and super block, data block and boot block. Case study of Unix file system

Mapping of Course Outcomes for Unit VI	CO6

UNIT VI	I/O System	06 Hours

I/O Management: I/O devices, Organization of I/O functions, Design issues related to I/O devices, I/O Buffering. ***Disk Scheduling***- FCFS, SCAN, SSTF, LOOK

Books and Other Resources

Text Books:

1. " Abranhan Silberschatz, Peter B Galvin, Greg Gagne; Operating System Concepts, Wiley India Students Edition, 8th Edition, ISBN: 978-81-265-2051-0

2. Andrew S. Tanenbaum; Modern Operating Systems; Prentice Hall of India Publication; 3rd Edition. ISBN: 978-81-203-3904-0

Reference Books:

1. Milan Milenkovic; Operating Systems; Tata McGraw Hill; Second Edition. ISBN: 0-07-044700-4
2. Maurice J. Bach; The Design of the Unix Opearating System; Prentice Hall of India; ISBN: 978-81-203-0516-8
3. Uresh Vahalia; Unix Internals, The New Frontiers; Prentice Hall; ISBN: 0-13-101908-2

Guidelines for Term Work Assessment:

Term work assessment will be based on overall performance of Laboratory assignments performed by students.

Guidelines for Practical Examination :

Internal and External Examine will take jointly oral examination of not more than 2 students at a time.

Understanding of the fundamentals, effective and efficient implementation.

Guidelines for Laboratory Conduction :

Operating System recommended :- 64-bit Open source Linux or its derivative

Programming tools recommended: - C++/ Java/ Python

Practical Assignments

1. Demonstration of Installation of Linux Operating System and Exploration of Unix/Linux Commands (File, Directory and Process commands).
2. Write a program to implement operations on processes using fork and join system calls.
3. Simulation of the scheduling algorithms. For example: First Come First Serve (FCFS), Shortest Remaining Time Next (SRTN).
4. Simulation of scheduling algorithms. For example: Round-Robin (RR), Pre-emptive Priority scheduling.
5. Write a program to implement Reader-Writer problem using semaphores
6. Write a program to implement Producer-Consumer problem
7. Write a program to implement Banker's Algorithm for deadlock handling
8. Simulation of Page replacement algorithms. For example: First-In-First-Out, Least Recently Used, optimal page replacement.
9. Simulation of memory allocation strategies. For example: First Fit, Best Fit and Worst Fit.

CO25204: Community Engagement Project

Teaching Scheme:	Credit: 02	Examination Scheme:
PR:04 Hrs/Week		Termwork:40 Marks Oral : 30 Marks

Prerequisite: Students should have prior knowledge of

1. Basic understanding of social and ethical responsibilities
2. Teamwork and communication skills acquired in prior coursework or group activities
3. Familiarity with problem-solving methodologies and project planning
4. Conversation in local language

Companion Course :

1. CEP is an experiential learning approach that combines education, learning, community devel-opment, and meaningful community service.
2. The project involves students in community development and service activities and applies the experience to personal and academic development.
3. The targeted contribution of college students to the village/local development will benefit the community.
4. The college has an opportunity to help students become more socially conscious and responsible while simultaneously becoming a socially conscious organization.

Course Objective:

1. To understand social and technological problems in local communities.
2. To develop awareness about the application of computer engineering in society.
3. To provide practical exposure through community interaction and fieldwork.
4. To develop solutions using appropriate computer engineering technologies.
5. To promote teamwork, leadership, and social responsibility.

Course Outcomes:

On completion of this course students will be able to

1. Identify community issues and formulate appropriate project objectives.
2. Conduct community surveys and analyze the requirements of beneficiaries.
3. Design suitable technology-based solutions for identified problems.
4. Implement and demonstrate solutions using appropriate computer engineering technologies.
5. Evaluate the effectiveness and social impact of implemented solutions.

Course Contents

Preamble:

Community Engagement Project (CEP) is an experiential learning approach designed to provide students with opportunities to develop technical knowledge, practical skills, social responsibility, and problem-solving abilities through active engagement with local communities. It enables students to identify, understand, and address authentic social, technological, and environmental challenges faced by communities by applying the concepts and technologies of computer engineering.

The Community Engagement Project goes beyond conventional classroom learning by encouraging students to interact with community members, identify real-life problems, analyze community needs and develop

appropriate technology-based solutions or conduct socially beneficial activities. It emphasizes learning by doing through meaningful participation, collaboration and practical implementation.

Community Engagement Project (CEP) requires regular mentoring and guidance by faculty throughout the semester for the successful completion of activities and project tasks undertaken by students. Students shall work in batches, with each batch divided into sub-groups of 3 to 4 students. Each group shall identify a relevant community problem and surveys, interact with beneficiaries, and undertake suitable activities such as digital literacy, cybersecurity awareness, technology-based solutions, social awareness programmers or community-oriented applications.

The selected community engagement activities and project work shall be carried out throughout the semester under the supervision of assigned faculty mentors. Faculty members involved in CEP shall provide regular guidance, monitor student progress and evaluate project outcomes. Credits for Community Engagement Project shall be awarded based on continuous internal assessment, active community participation, project implementation, documentation, presentation and evaluation at the end of the semester.

Group Structure:
Working in supervisor/mentor monitored groups; the students plan, manage, and complete a task/project/activity which addresses the stated problem.

1. There should be a team/group of 3-4 students
2. A supervisor/mentor teacher assigned to individual groups

Selection of Community Engagement Project / Problem:
The community-based project-oriented learning model is recommended for the Community Engagement Project (CEP). The process begins with identifying real-life problems or challenges faced by local communities, villages, schools, NGOs or other social organizations. Students shall interact with community members, conduct surveys and identify problems that can be addressed through the application of Computer engineering knowledge and technologies.

The identified problems may be social, educational, environmental, technical, agricultural or healthcare-related. Students shall analyse the selected problem and propose suitable solutions using Computer engineering concepts such as programming, web development, mobile applications, database management, Artificial intelligence, IoT and cybersecurity.

The selected project should address a genuine community need and provide practical social benefits. An interdisciplinary approach may be adopted wherever required for analysing and solving the identified problems.

Students are encouraged to use appropriate technologies to investigate community issues, collaborate with stakeholders, analyse collected data, develop solutions and present their outcomes. Activities may include solving real-life community problems, conducting digital literacy and cybersecurity awareness programmes, developing community-oriented software applications, field visits, surveys, case studies and preparation of project reports.

Suggestive list of topics under Community Engagement Project

Community Engagement Projects should focus on specific local issues in the students' nearby areas, and students are expected to carry them out with involvement, commitment, responsibility and accountability.

Mentors may assign a student or a group of students any of the following or any **Community problems**

Health and Wellness

- Health and hygiene of school students, homemakers and the elderly
- Health interventions and awareness programmes
- Healthcare awareness programmes and their impact
- Traditional and modern healthcare methods
- Incidence of diabetes and other chronic diseases
- Blood groups and blood levels
- Yoga awareness and practice

Food and Nutrition

- Food habits
- Herbal remedies and nutrition
- Food adulteration
- Chemicals in daily life

Environment and Agriculture

- Water facilities and drinking water availability
- Air, noise and water pollution
- Plantation and soil conservation
- Horticulture
- Organic farming
- Renewable energy and solar systems

Youth and Society

- Use and misuse of mobile phones
- Career orientation of youth
- Women's education and empowerment
- Music and dance

Guidelines for TW Assessment (40 Marks)

1. Problem Identification and Research (7 Marks)

- 4 Marks: Clear, relevant problem statement supported by surveys and literature.
- 3 Marks: Effective research, including community input and background information.

2. Design and Concept Development (7 Marks)

- 4 Marks: Use of UML/ER/DFD design tools for concept visualization.
- 3 Marks: Practicality in terms of cost effective feasibility and sustainability.

3. Prototyping and Testing (7 Marks)

- 4 Marks: Testing methods, results, and prototype improvements.
- 3 Marks: Documentation of prototyping process (photos, notes).

4. Implementation and Community Engagement (7 Marks)

- 4 Marks: Successful deployment with community interaction, training, and feedback.
- 3 Marks: Modifications based on community feedback.

5. Documentation , Report Writing and IPR/Publication (7 Marks)

- 4 Marks: Clear, organized, and accurate project report with proper citations
- 3 Mark: IPR/Publications.

6. Presentation and Communication (5 Marks)

- 3 Marks: Clear and engaging presentation to stakeholders

Guidelines for OR Assessment (30 Marks)

1. Presentation Structure (10 Marks)

- Clarity & Organization (5 Marks): Clear structure, logical flow from problem to solution.
- Logical Flow (5 Marks): Well-organized content, smooth progression through key points.

2. Technical Understanding (8 Marks)

- Engineering Concepts (4 Marks): Demonstrates deep understanding of applied technology and engineering principles.
- Problem-Solving (4 Marks): Clear explanation of solution and engineering approach.

3. Design & Prototyping (6 Marks)

- Design Explanation (3 Marks): Clear explanation of software design process.
- Prototype/Model Explanation (3 Marks): Functionality and testing outcomes of the prototype.

4. Community Engagement (4 Marks)

- Community Involvement (2 Marks): Explains community feedback and participation.
- Impact (2 Marks): Describes the solution's impact on the community.

5. Response to Questions (2 Marks)

- Confidence & Accuracy (2 Marks): Clear, confident, and accurate answers

Prof. D.M.D.Shelar

Dr. G.I. Chhajed

HOD-Comp. Dr. A.M. Jagtap

Course Name with Code: Environmental Studies (HS25211TH)		
Teaching Scheme: **TH : 2 Hrs./week**	**Credits** **02**	**Examination Scheme:** **Activity : 10 Marks** **End Semester : 60 Marks**

Prerequisite:

Basic concepts of biodiversity and environmental issues.

Course Objectives:

1. Understand the fundamental concepts of environmental science and its relevance to engineering.
2. Analyze the environmental impact of various engineering industries.
3. Learn about sustainable engineering practices, pollution control, and waste management.
4. Study environmental laws in India and global initiatives for environmental conservation.

Course Outcomes:

On completion of the course, learner will be able to:

CO-1: Understand the components of the environment and types of energy resources.
CO-2: Analyze the impact of engineering industries on the environment.
CO-3: Learn sustainable engineering solutions for mitigating environmental damage.
CO-4: Aware of Indian and global initiatives for environmental protection.

Course Contents

Unit I: Introduction to environmental studies **(06 Hours)**

Importance of Environmental Studies, Components of the Environment: Atmosphere, Hydrosphere, Lithosphere, and Biosphere, Ecosystems and Biodiversity: Types, Importance, and Conservation, Sustainable Development Goals (SDGs) and Role of Engineers in Sustainability, Renewable and Non-Renewable Resources, Water Resources: Overuse, Pollution, and Engineering Solutions, Energy Resources: Fossil Fuels, Nuclear Power, and Renewable Energy Alternatives, Land Resources: Soil Degradation, Deforestation, and Urbanization.

Unit II: Impact of Engineering Industries on Environment **(07 Hours)**

Manufacturing & Automobile Industry: Air pollution, Carbon emissions, Waste disposal, Chemical & Pharmaceutical Industry: Water and soil contamination, Hazardous waste, Construction & Infrastructure: Land degradation, Dust pollution, Waste generation, Electronics & IT Industry: E-waste, Energy consumption, Semiconductor waste, Power Generation (Thermal, Hydropower, Nuclear): Pollution, Waste heat, Radiation hazards, Causes and Effects of Climate Change, Global Warming and Greenhouse Effect.

Unit III: Engineering Solutions for Environmental Mitigation and Sustainable Practices (07 Hours)

Carbon Capture and Storage (CCS), Eco-friendly Materials, Sustainable Design & Life Cycle Assessment (LCA), Energy-efficient Technologies & Smart Grids, Case Studies on Successful Pollution Reduction Waste Management Strategies: Solid Waste and Biomedical Waste Management, E-Waste: Sources, Impact, and Recycling, Hazardous Waste Handling and Treatment, Circular Economy and Zero-Waste Technologies Sustainable Engineering Practices: Renewable Energy Technologies (Solar, Wind, Biomass, Hydropower) Green Buildings and Sustainable Architecture, Electric Vehicles and Smart Transportation Systems, Sustainable Agriculture and Water Conservation Technologies.

Unit IV: Environmental Initiatives in India and Worldwide **(06 Hours)**

National Initiatives: Swachh Bharat Abhiyan, Namami Gange, National Green Tribunal (NGT), Corporate Social Responsibility (CSR) & Environmental Compliance, Environmental Activism and the Role of NGOs, Environmental Laws and Policies in India, The Environmental Protection Act, 1986, Role of Central Pollution Control Board (CPCB) and State Pollution Control Boards (SPCB), International Environmental Agreements (Kyoto Protocol, Paris Agreement, COP Summits), Global Initiatives: UNEP, IPCC, World Bank Environmental Policies.

VPKBIET BARAMATI
Affiliated to S.P.P.U. Pune Id No. PU/PN/Engg/152/2000
PUNE (MS.)

Books & Other Resources:

Text books:

1. Benny Joseph, Environmental Studies, McGraw Hill Education, 3rd Edition, 2021. Anubha Kaushik & C.P. Kaushik, Environmental Studies, New Age International Publishers, 5th Edition, 2022.

Reference books:

1. R. Rajagopalan, Environmental Studies: From Crisis to Cure, Oxford University Press, 3rd Edition, 2021.
2. Erach Bharucha, Textbook of Environmental Studies for Undergraduate Courses, University Press, 3rd Edition, 2021.
3. Suresh K. Dhameja, Environmental Science and Engineering, S.K. Kataria & Sons, 2nd Edition, 2020.

Additional Reports & Resources:

- Government of India Ministry of Environment, Forest & Climate Change (MoEFCC) Reports (Website).
- United Nations Environment Programme (UNEP) Reports (Website).
- IPCC Climate Change Reports (Website).
- Central Pollution Board (CPCB) Reports (Website).

Activity:

Perform any two activities of the following.

1. Assignment on each unit.
2. Poster Presentation.
3. Seminar Presentation.
4. Case study

Mr. A.M. Gaikwad	Mrs. Gauri Bhoite	Dr. S.M. Bhosale	Dr. S.B. Kandal
Course Coordinator	BOS, HSSM	Dean Academics	Principal

Principal
Vidya Pratishthan's
Kamalnayan Bajaj Institute of
Engineering & Technology, Baramati
Vidyanagari, Baramati-413133

V.P.K.B.I.E.T. BARAMATI
Affiliated to S.P.P.U. Pune
Id No./PU/PN/Engg/152/2000
(MS.)