

**w.e.f.2023-24**



**DIET23**

**B. Tech.**

**COMPUTER SCIENCE AND ENGINEERING- ARTIFICIAL INTELLIGENCE**

**(B.Tech1st Year Syllabus)**

**Department of Computer Science and Engineering- Artificial Intelligence**

**(B.Tech. CSE-AI & ML)**

****

**DHANEKULA INSTITUTE OF**

**ENGINEERING AND TECHNOLOGY**

**(Approved by AICTE, Accredited by NBA, Affiliated to JNTUK, Kakinada)**

**Ganguru, Vijayawada**

**AndhraPradesh-521139,**

**INDIA.**

[**www.diet.ac.in**](http://www.diet.ac.in)



**DHANEKULA INSTITUTE OF ENGINEERING & TECHNOLOGY**

Department of Computer Science & Engineering-AI&ML

**VISION – MISSION - PEOs**

|  |  |
| --- | --- |
| **Institute Vision** | Pioneering Professional Education through Quality |
| **Institute Mission** | Providing Quality Education through state-of-art infrastructure, laboratories and committed staff.  Moulding Students as proficient, competent, and socially responsible engineering personnel with ingenious intellect.  Involving faculty members and students in research and development works for betterment of society. |
| **Department Vision** | To empower students of Computer Science and Engineering Department to be technologically adept, innovative, global citizens possessing human values. |
| **Department Mission** | To Encourage students to become self-motivated and problem-solving individuals.  To prepare students for professional career with academic excellence and leadership skills.  To Empower the rural youth with computer education.  To Create Centre’s of excellence in Computer Science and Engineering. |
| **Program Educational Objectives(PEOs)** | Graduates of Computer Science & Engineering will:  **PEO1:** Excel in Professional career through knowledge in mathematics and engineering principles.  **PEO2:** Able to pursue higher education and research.  **PEO3:** Communicate effectively, recognize, and incorporate societal needs in their professional endeavors.  **PEO4:** Adapt to technological advancements by continuous learning. |



**DHANEKULA INSTITUTE OF ENGINEERING & TECHNOLOGY**

**Department of Computer Science & Engineering-AI&ML**

**POs/PSOs**

**List Program Outcomes**

|  |  |
| --- | --- |
| **1** | **Engineering Knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals and an engineering specialization to the solution of complex engineering problems. |
| **2** | **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. |
| **3** | **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 |
| **4** | **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. |
| **5** | **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. |
| **6** | **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. |
| **7** | **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. |
| **8** | **Ethics**: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice. |
| **9** | **Individual and team work**: Function effectively as an individual, and as a member or leader in diverse teams, and in multi disciplinary settings. |
| **10** | **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. |
| **11** | **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. |
| **12** | **Life-long 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. |

**List Program Specific Outcomes**

**PSO1:** Have expertise in algorithms, networking, web applications and software engineering for efficient design of computer-based systems of varying complexity.

**PSO2:** Qualify in national international level competitive examinations for successful higher studies and employment.

**B.Tech.–II Year I Semester**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **S.No.** | **Category** | **Title** | **L/D** | **T** | **P** | **Credits** |
| 1 | BS&H | Discrete Mathematics & Graph  Theory | 3 | 0 | 0 | 3 |
| 2 | BS&H | Universal human values – understanding harmony and  Ethical human conduct | 2 | 1 | 0 | 3 |
| 3 | Engineering  Science | Artificial Intelligence | 3 | 0 | 0 | 3 |
| 4 | Professional Core | Advanced Data Structures&  Algorithms Analysis | 3 | 0 | 0 | 3 |
| 5 | Professional Core | Object Oriented Programming  Through Java | 3 | 0 | 0 | 3 |
| 6 | Professional Core | Advanced Data Structures and  Algorithms Lab | 0 | 0 | 3 | 1.5 |
| 7 | Professional Core | Object Oriented Programming  Through Java Lab | 0 | 0 | 3 | 1.5 |
| 8 | Skill Enhancement  course | Python programming | 0 | 1 | 2 | 2 |
| 9 | Audit Course | Environmental Science | 2 | 0 | 0 | - |
| 10 | B S & H | Health and wellness, Yoga and Sports | - | - | 1 | 0.5 |
| **Total** | | | **16** | **2** | **8** | **20** |

**B.Tech.–II Year II Semester**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **S.No.** | **Category** | **Title** | **L/D** | **T** | **P** | **Credits** |
| 1 | Management Course-I | Optimization Techniques | 2 | 0 | 0 | 2 |
| 2 | Engineering Science/Basic  Science | Probability & Statistics | 3 | 0 | 0 | 3 |
| 3 | Professional Core | Machine Learning | 3 | 0 | 0 | 3 |
| 4 | Professional Core | Database Management  Systems | 3 | 0 | 0 | 3 |
| 5 | Professional Core | Digital Logic & Computer  Organization | 3 | 0 | 0 | 3 |
| 6 | Professional Core | Machine Learning Lab | 0 | 0 | 3 | 1.5 |
| 7 | Professional Core | Database Management  Systems Lab | 0 | 0 | 3 | 1.5 |
| 8 | Skill Enhancement course | FullStackdevelopment-1 | 0 | 1 | 2 | 2 |
| 9 | BS&H | Design Thinking  & Innovation | 1 | 0 | 2 | 2 |
| **Total** | | | **15** | **01** | **12** | **20** |
| Mandatory Community Service Project Internship of 08 weeks duration during summer  vacation | | | | | | |

**II- I Semester**

****

Regulation

**DIET23**

**ARTIFICIAL INTELLIGENCE**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Course Category: | **Engineering Science (ES)** | Credits: 3 | | | |
| Course Type: | **Theory** | Lecture-Tutorial-Practice: | 3 | 0 | 0 |
| Prerequisites | Mathematical Foundations of Computer Science, Background in linear algebra, data structures and algorithms. | Continuous Evaluation | 30M | | |
| Semester End Evaluation | 70M | | |
| Total Marks | 100M | | |

**Course Outcomes:**

|  |  |  |
| --- | --- | --- |
| After Successful Completion of course, the student will be able to: | | |
| CO No: | Course Outcome Description | K - Level |
| CO1 | Apply the principles of problem solving agents to formulate and structure problems | Applying |
| CO2 | Build various algorithms to solve searching and gamming techniques. | Applying |
| CO3 | Make use various methods of knowledge representation like predicate logic, rule based deduction systems. | Applying |
| CO4 | Construct various learning methods using logic concepts. | Applying |
| CO5 | Compare different types of expert systems. | Analyzing |

Note: K-Level is defined From Blooms Taxonomy

Contribution of Course Outcomes mapping with POs & PSOs (**1**- Low, **2** – Moderate,**3** – High)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **CO No.** | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** | **PSO1** | **PSO2** |
| CO1 | 3 | - | - | - | - | - | - | - | - | - | - | - | 3 | 3 |
| CO2 | 3 | - | - | - | - | - | - | - | - | - | - | - | 3 | 3 |
| CO3 | 3 | - | - | - | - | - | - | - | - | - | - | - | 3 | 3 |
| CO4 | 3 | - | - | - | - | - | - | - | - | - | - | - | 3 | 3 |
| CO5 | - | 3 | - | - | - | - | - | - | - | - | - | - | 3 | 3 |

**COURSE CONTENT:**

**UNIT – I**

Introduction: AI problems, foundation of AI and history of AI intelligent agents: Agents and Environments, the concept of rationality, the nature of environments, structure of agents, problem solving agents, problem formulation.

**UNIT - II**

Searching- Searching for solutions, uniformed search strategies – Breadth first search, depth first Search. Search with partial information (Heuristic search) Hill climbing, A\* ,AO\* Algorithms, Problem reduction, Game Playing-Adversial search, Games, mini-max algorithm, optimal decisions in multiplayer games, Problem in Game playing, Alpha-Beta pruning, Evaluation functions.

**UNIT - III**

Representation of Knowledge: Knowledge representation issues, predicate logic- logic programming, semantic nets- frames and inheritance, constraint propagation, representing knowledge using rules, rules based deduction systems. Reasoning under uncertainty, review of probability. Bayes’ probabilistic interferences and dempstershafer theory.

**UNIT - IV**

Logic concepts: First order logic. Inference in first order logic, propositional vs. first order inference, unification & lifts forward chaining, Backward chaining, Resolution, Learning from observation Inductive learning, Decision trees, Explanation based learning, Statistical Learning methods, Reinforcement Learning.

**UNIT – V**

Expert Systems: Architecture of expert systems, Roles of expert systems – Knowledge Acquisition Meta knowledge Heuristics. Typical expert systems – MYCIN, DART, XCON: Expert systems shells.

**Textbooks:**

1. S. Russel and P. Norvig, “Artificial Intelligence – A Modern Approach”, SecondEdition, Pearson Education.(2019)

2. Kevin Night and Elaine Rich, Nair B., “Artificial Intelligence (SIE)”, Mc Graw Hill(2017)

**Reference Books:**

1. David Poole, Alan Mackworth, Randy Goebel,”Computational Intelligence: a logical approach”, Oxford University Press.

2. G. Luger, “Artificial Intelligence: Structures and Strategies for complex problemsolving”, Fourth Edition, Pearson Education.

3. J. Nilsson, “Artificial Intelligence: A new Synthesis”, Elsevier Publishers.

4. Artificial Intelligence, SarojKaushik, CENGAGE Learning.

**E-Resources:**

1. <https://ai.google/>

2. https://swayam.gov.in/nd1\_noc19\_me71/preview

Course

****

Regulation

**D23**

**ADVANCED DATA STRUCTURES & ALGORITHM ANALYSIS**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| CourseCategory: | Professional Core | Credits: 3 | | | |
| Course Type: | **Theory** / Tutorial /Practical | Lecture-Tutorial-Practice: | 3 | 0 | 0 |
| Prerequisites | * Sound knowledge of basic data structures and implementations * Basics of Mathematics and Programming. | Continuous Evaluation | 30M | | |
| Semester End Evaluation | 70M | | |
| Total Marks | 100M | | |

**Course Outcomes:**

|  |  |  |
| --- | --- | --- |
| After Successful Completion of course,the student will be able to: | | |
| CO No: | Course Outcome Description | K - Level |
| CO1 | Illustrate AVL, B-Trees with insights into Algorithm Analysis with Asymptotic Notations | Applying |
| CO2 | Examine Heap Trees, Graphs and solve various applications using divide and conquer technique | Analyzing |
| CO3 | Analyze different applications using Greedy Method and Dynamic Programming | Analyzing |
| CO4 | Examine Backtracking, Branch and Bound to solve various problems | Analyzing |
| CO5 | Compare NP Hard and NP Complete Problems using different problems | Analyzing |

Note: K-Level is defined From Blooms Taxonomy

Contribution of Course Outcomes mapping with POs & PSOs (**1**- Low, **2** – Moderate,**3** – High)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **CO No.** | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** | **PSO1** | **PSO2** |
| CO1 | 3 | - | - | - | - | - | - | - | - | - | - | - | 3 | 3 |
| CO2 | - | 3 | - | - | - | - | - | - | - | - | - | - | 3 | 3 |
| CO3 | - | 3 | - | - | - | - | - | - | - | - | - | - | 3 | 3 |
| CO4 | - | 3 | - | - | - | - | - | - | - | - | - | - | 3 | 3 |
| CO5 | - | 3 | - | - | - | - | - | - | - | - | - | - | 3 | 3 |

**COURSE CONTENT:**

UNIT-1:

Introduction to Algorithm Analysis, Space and Time Complexity analysis, Asymptotic Notations.(already stuided in 1-2-Data Strructures subject)

AVL Trees – Creation, Insertion, Deletion operations and Applications

B-Trees – Creation, Insertion, Deletion operations and Applications

UNIT-2:

Heap Trees (Priority Queues) – Min and Max Heaps, Operations and Applications

Graphs – Terminology, Representations, Basic Search and Traversals, ConnectedComponents and Biconnected Components, applications

Divide and Conquer: The General Method, Quick Sort, Merge Sort, Strassen’s matrixmultiplication, Convex HullNotations.(involves only mathematical derivation no performance analysis)

UNIT-3:

Greedy Method: General Method, Job Sequencing with deadlines, Knapsack Problem,Minimum cost spanning trees, Single Source Shortest Paths

Dynamic Programming: General Method, All pairs shortest paths, Single Source ShortestPaths– General Weights (Bellman Ford Algorithm), Optimal Binary Search Trees, 0/1Knapsack, String Editing, Travelling Salesperson problem

UNIT-4:

Backtracking: General Method, 8-Queens Problem, Sum of Subsets problem, GraphColoring, 0/1 Knapsack Problem

Branch and Bound: The General Method, 0/1 Knapsack Problem, Travelling Salesperson problem

UNIT-5:

NP Hard and NP Complete Problems: Basic Concepts, Cook’s theorem

NP Hard Graph Problems: Clique Decision Problem (CDP), Chromatic Number DecisionProblem (CNDP), Traveling Salesperson Decision Problem (TSP)

NP Hard Scheduling Problems: Scheduling Identical Processors, Job Shop Scheduling (heavy to cover)

**TEXTBOOKS:**

1. Fundamentals of Data Structures in C++, HorowitzEllis, SahniSartaj, MehtaDinesh,

2ndEdition, Universities Press, 2008.

1. Computer Algorithms in C++, Ellis Horowitz, SartajSahni, SanguthevarRajasekaran,

2nd Edition, University Press, 2008.

**REFERENCE BOOKS:**

1. Data Structures and program design in C, Robert Kruse, 2nd Edition, Pearson Education Asia, 2006
2. An introduction to Data Structures with applications, Trembley& Sorenson, McGrawHill, 2nd edition, 2017

**E-RESOURCES/DIGITAL MATERIAL:**

1. <https://www.tutorialspoint.com/advanced_data_structures/index.asp>
2. <http://peterindia.net/Algorithms.html>
3. Abdul Bari,Introduction to Algorithms (youtube.com)
4. Introduction to the NPTEL MOOC on Design and Analysis of Algorithms by Prof. MUKUND MADHAVAN, IIT MADRAS,[Design and Analysis of Algorithms](Design%20and%20Analysis%20of%20Algorithms)<https://archive.nptel.ac.in/courses/106/106/106106131/>

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| CourseCategory: | Professional Core | Credits : 3 | | | |
| Course Type: | Theory / Tutorial /Practical | Lecture-Tutorial-Practice: | 3 | 0 | 0 |
| Prerequisites | Basic Programming | Continuous Evaluation | 30M | | |
| Semester End Evaluation | 70M | | |
| Total Marks | 100M | | |

**Course Outcomes:**

|  |  |  |
| --- | --- | --- |
| After Successful Completion of course, the student will be able to: | | |
| CO No: | Course Outcome Description | K - Level |
| CO1 | Apply object oriented concepts, control structures in Java. | Applying |
| CO2 | Applying Object oriented constructs such as various class hierarchies, Methods | Applying |
| CO3 | Applying concepts like arrays, inheritances, interfaces in java | Applying |
| CO4 | Applying packages and exception handling in java | Applying |
| CO5 | Apply Multi threading ,string methods and JDBC connections, Java FX in java | Applying |

Note: K-Level is defined From Blooms Taxonomy

Contribution of Course Outcomes mapping with POs & PSOs (**1**- Low, **2** – Moderate,**3** – High)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **CO No.** | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** | **PSO1** | **PSO2** |
| CO1 | 3 | - | - | - | - | - | - | - | - | - | - | - | 3 | 3 |
| CO2 | 3 | - | - | - | - | - | - | - | - | - | - | - | 3 | 3 |
| CO3 | 3 | - | - | - | - | - | - | - | - | - | - | - | 3 | 3 |
| CO4 | 3 | - | - | - | - | - | - | - | - | - | - | - | 3 | 3 |
| CO5 | 3 | - | - | - | - | - | - | - | - | - | - | - | 3 | 3 |

**COURSE CONTENT:**

**UNIT-1:**

**Object Oriented Programming**: Basic concepts, Principles,

Program Structure in Java: Introduction, Writing Simple Java Programs, Elements or Tokens in Java Programs, Java Statements, Command Line Arguments, User Input to Programs, Escape Sequences Comments, Programming Style.

**Data Types**, Variables, and Operators :Introduction, Data Types in Java, Declaration of Variables, Data Types, Type Casting, Scope of Variable Identifier, Literal Constants, Symbolic Constants, Formatted Output with printf() Method, Static Variables and Methods, Attribute Final, Introduction to Operators, Precedence and Associativity of Operators, Assignment Operator ( = ), Basic Arithmetic Operators, Increment (++) and Decrement (- -) Operators, Ternary Operator, Relational Operators, Boolean Logical Operators, Bitwise Logical Operators.

**Control Statements**: Introduction, if Expression, Nested if Expressions, if–else Expressions, Ternary Operator?:, Switch Statement, Iteration Statements, while Expression, do–while Loop, for Loop, Nested for Loop, For–Each for Loop, Break Statement, Continue Statement.

**UNIT-2:**

**Classes and Objects:** Introduction,ClassDeclarationandModifiers,ClassMembers,Declaration of Class Objects, Assigning One Object to Another, Access Control for ClassMembers, Accessing Private Members of Class, Constructor Methods for Class, OverloadedConstructor Methods, Nested Classes, Final Class and Methods, Passing Arguments by Value and by Reference, Keyword this.

**Methods**: Introduction, Defining Methods, Overloaded Methods, Overloaded Constructor Methods, Class Objects as Parameters in Methods, Access Control, Recursive Methods, Nesting of Methods, Overriding Methods, Attributes Final and Static

**Java I/O and File**: Java I/O API, standard I/O streams, types, Byte streams, Character streams, Scanner class, Files in Java(Text Book 2)

**UNIT-3:**

**Arrays**: Introduction, Declaration and Initialization of Arrays, Storage of Array in Computer Memory, Accessing Elements of Arrays, **Operations on Array Elements**, Assigning Array to Another Array, Dynamic Change of Array Size, Sorting of Arrays, Search for Values in Arrays, Class Arrays, Two-dimensional Arrays, Arrays of Varying Lengths, Three- dimensional Arrays, **Arrays as Vectors.**

**Inheritance**: Introduction, Process of Inheritance, Types of Inheritances, Universal Super Class-Object Class, Inhibiting Inheritance of Class Using Final, Access Control and Inheritance, Multilevel Inheritance, Application of Keyword Super, Constructor Method and Inheritance, Method Overriding, Dynamic Method Dispatch, Abstract Classes, Interfaces and Inheritance. **Interfaces**: Introduction, Declaration of Interface, Implementation of Interface, Multiple Interfaces, Nested Interfaces, Inheritance of Interfaces, Default Methods in Interfaces, Static Methods in Interface, Functional Interfaces, Annotations.

**UNIT-4:**

**Packages and Java Library**: Introduction, Defining Package, Importing Packages and Classes into Programs, Path and Class Path, Access Control, Packages in Java SE, Java.lang Package and its Classes, Class Object, Enumeration, class Math, Wrapper Classes, Auto-boxing and Auto-unboxing, Java util Classes and Interfaces, Formatter Class, Random Class, Time Package, Class Instant (java.time.Instant), Formatting for Date/Time in Java, **Temporal Adjusters Class, Temporal Adjusters Class.**

**Exception Handling**: Introduction, Hierarchy of Standard Exception Classes, Keywords throws and throw, try, catch, and finally Blocks, Multiple Catch Clauses, Class Throwable, Unchecked Exceptions, Checked Exception

**String Handling in Java:** Introduction, Interface Char Sequence, Class String, Methods for Extracting Characters from Strings, Comparison, Modifying, Searching; Class String Buffer, String Builder

**UNIT-5:**

**Multithreaded Programming**: Introduction, Need for Multiple Threads Multi threaded Programming for Multi-core Processor, Thread Class, Main Thread-Creation of New Threads, Thread States, Thread PrioritySynchronization, Deadlock and Race Situations, Inter-thread Communication - Suspending, Resuming, and Stopping of Threads.

**Java Database Connectivity**: Introduction, JDBC Architecture, Installing MySQL and MySQL Connector/J, JDBC Environment Setup, Establishing JDBC Database Connections, Result Set Interface

**Java FX GUI**: Java FX Scene Builder, Java FX App Window Structure, displaying text and image, event handling, laying out nodes in scene graph, mouse events (Text Book 3)

**TEXT BOOKS:**

1. JAVA One Step Ahead, Anitha Seth, B.L.Juneja, Oxford.- **2017**
2. Joy with JAVA, Fundamentals of Object Oriented Programming, Debasis Samanta, Monalisa Sarma, Cambridge, -**2023.**
3. JAVA 9 for Programmers, Paul Deitel, Harvey Deitel, 4th Edition, Pearson. **2014**

**REFERENCE BOOKS:**

(i) The complete Reference Java, 11thedition, Herbert Schildt,TMH

(ii)Introduction to Java programming, 7th Edition, Y Daniel Liang, Pearson

**E-RESOURCES/DIGITAL MATERIAL:**

(a)https://nptel.ac.in/courses/106/105/106105191/

(b)https://infyspringboard.onwingspan.com/web/en/app/toc/lex\_auth\_012880464547618816347\_shared/overview

**OBJECT ORIENTED PROGRAMMING THROUGH JAVA**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| CourseCategory: | Professional Core | Credits : 3 | | | |
| Course Type: | Theory / Tutorial /Practical | Lecture-Tutorial-Practice: | 3 | 0 | 0 |
| Prerequisites | Basic Programming | Continuous Evaluation | 30M | | |
| Semester End Evaluation | 70M | | |
| Total Marks | 100M | | |

**Course Outcomes:**

|  |  |  |
| --- | --- | --- |
| After Successful Completion of course, the student will be able to: | | |
| CO No: | Course Outcome Description | K - Level |
| CO1 | Apply object oriented concepts, control structures in Java. | Applying |
| CO2 | Applying Object oriented constructs such as various class hierarchies, Methods | Applying |
| CO3 | Applying concepts like arrays, inheritances, interfaces in java | Applying |
| CO4 | Applying packages and exception handling in java | Applying |
| CO5 | Apply Multi threading ,string methods and JDBC connections, Java FX in java | Applying |

Note: K-Level is defined From Blooms Taxonomy

Contribution of Course Outcomes mapping with POs & PSOs (**1**- Low, **2** – Moderate,**3** – High)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **CO No.** | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** | **PSO1** | **PSO2** |
| CO1 | 3 | - | - | - | - | - | - | - | - | - | - | - | 3 | 3 |
| CO2 | 3 | - | - | - | - | - | - | - | - | - | - | - | 3 | 3 |
| CO3 | 3 | - | - | - | - | - | - | - | - | - | - | - | 3 | 3 |
| CO4 | 3 | - | - | - | - | - | - | - | - | - | - | - | 3 | 3 |
| CO5 | 3 | - | - | - | - | - | - | - | - | - | - | - | 3 | 3 |

**COURSE CONTENT:**

**UNIT-1:**

**Object Oriented Programming**: Basic concepts, Principles,

Program Structure in Java: Introduction, Writing Simple Java Programs, Elements or Tokens in Java Programs, Java Statements, Command Line Arguments, User Input to Programs, Escape Sequences Comments, Programming Style.

**Data Types**, Variables, and Operators :Introduction, Data Types in Java, Declaration of Variables, Data Types, Type Casting, Scope of Variable Identifier, Literal Constants, Symbolic Constants, Formatted Output with printf() Method, Static Variables and Methods, Attribute Final, Introduction to Operators, Precedence and Associativity of Operators, Assignment Operator ( = ), Basic Arithmetic Operators, Increment (++) and Decrement (- -) Operators, Ternary Operator, Relational Operators, Boolean Logical Operators, Bitwise Logical Operators.

**Control Statements**: Introduction, if Expression, Nested if Expressions, if–else Expressions, Ternary Operator?:, Switch Statement, Iteration Statements, while Expression, do–while Loop, for Loop, Nested for Loop, For–Each for Loop, Break Statement, Continue Statement.

**UNIT-2:**

**Classes and Objects:** Introduction,ClassDeclarationandModifiers,ClassMembers,Declaration of Class Objects, Assigning One Object to Another, Access Control for ClassMembers, Accessing Private Members of Class, Constructor Methods for Class, OverloadedConstructor Methods, Nested Classes, Final Class and Methods, Passing Arguments by Value and by Reference, Keyword this.

**Methods**: Introduction, Defining Methods, Overloaded Methods, Overloaded Constructor Methods, Class Objects as Parameters in Methods, Access Control, Recursive Methods, Nesting of Methods, Overriding Methods, Attributes Final and Static

**Java I/O and File**: Java I/O API, standard I/O streams, types, Byte streams, Character streams, Scanner class, Files in Java(Text Book 2)

**UNIT-3:**

**Arrays**: Introduction, Declaration and Initialization of Arrays, Storage of Array in Computer Memory, Accessing Elements of Arrays, **Operations on Array Elements**, Assigning Array to Another Array, Dynamic Change of Array Size, Sorting of Arrays, Search for Values in Arrays, Class Arrays, Two-dimensional Arrays, Arrays of Varying Lengths, Three- dimensional Arrays, **Arrays as Vectors.**

**Inheritance**: Introduction, Process of Inheritance, Types of Inheritances, Universal Super Class-Object Class, Inhibiting Inheritance of Class Using Final, Access Control and Inheritance, Multilevel Inheritance, Application of Keyword Super, Constructor Method and Inheritance, Method Overriding, Dynamic Method Dispatch, Abstract Classes, Interfaces and Inheritance. **Interfaces**: Introduction, Declaration of Interface, Implementation of Interface, Multiple Interfaces, Nested Interfaces, Inheritance of Interfaces, Default Methods in Interfaces, Static Methods in Interface, Functional Interfaces, Annotations.

**UNIT-4:**

**Packages and Java Library**: Introduction, Defining Package, Importing Packages and Classes into Programs, Path and Class Path, Access Control, Packages in Java SE, Java.lang Package and its Classes, Class Object, Enumeration, class Math, Wrapper Classes, Auto-boxing and Auto-unboxing, Java util Classes and Interfaces, Formatter Class, Random Class, Time Package, Class Instant (java.time.Instant), Formatting for Date/Time in Java, **Temporal Adjusters Class, Temporal Adjusters Class.**

**Exception Handling**: Introduction, Hierarchy of Standard Exception Classes, Keywords throws and throw, try, catch, and finally Blocks, Multiple Catch Clauses, Class Throwable, Unchecked Exceptions, Checked Exception

**String Handling in Java:** Introduction, Interface Char Sequence, Class String, Methods for Extracting Characters from Strings, Comparison, Modifying, Searching; Class String Buffer, String Builder

**UNIT-5:**

**Multithreaded Programming**: Introduction, Need for Multiple Threads Multi threaded Programming for Multi-core Processor, Thread Class, Main Thread-Creation of New Threads, Thread States, Thread PrioritySynchronization, Deadlock and Race Situations, Inter-thread Communication - Suspending, Resuming, and Stopping of Threads.

**Java Database Connectivity**: Introduction, JDBC Architecture, Installing MySQL and MySQL Connector/J, JDBC Environment Setup, Establishing JDBC Database Connections, Result Set Interface

**Java FX GUI**: Java FX Scene Builder, Java FX App Window Structure, displaying text and image, event handling, laying out nodes in scene graph, mouse events (Text Book 3)

**TEXT BOOKS:**

1. JAVA One Step Ahead, Anitha Seth, B.L.Juneja, Oxford.- **2017**
2. Joy with JAVA, Fundamentals of Object Oriented Programming, Debasis Samanta, Monalisa Sarma, Cambridge, -**2023.**
3. JAVA 9 for Programmers, Paul Deitel, Harvey Deitel, 4th Edition, Pearson. **2014**

**REFERENCE BOOKS:**

(i) The complete Reference Java, 11thedition, Herbert Schildt,TMH

(ii)Introduction to Java programming, 7th Edition, Y Daniel Liang, Pearson

**E-RESOURCES/DIGITAL MATERIAL:**

(a)https://nptel.ac.in/courses/106/105/106105191/

(b)https://infyspringboard.onwingspan.com/web/en/app/toc/lex\_auth\_012880464547618816347\_shared/overview

Course

****

Regulation

**D23**

**ADVANCED DATA STRUCTURES & ALGORITHM ANALYSIS LAB**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| CourseCategory: | Professional Core | Credits:1.5 | | | |
| Course Type: | Theory / Tutorial /**Practical** | Lecture-Tutorial-Practice: | 0 | 0 | 3 |
| Prerequisites | * Knowledge of implementation of basic data structures with arrays and linked lists. * Basics of Mathematics and Programming. | Continuous Evaluation | 30M | | |
| Semester End Evaluation | 70M | | |
| Total Marks | 100M | | |

**Course Outcomes:**

|  |  |  |
| --- | --- | --- |
| After Successful Completion of course,the student will be able to: | | |
| CO No: | Course Outcome Description | K - Level |
| CO1 | Implement data structures like AVL Trees, B-Tree, Heap Tree and Graphs | Applying |
| CO2 | Solve problems using different algorithmic approaches like Divide and Conquer, Greedy method, Dynamic Programming and Backtracking | Applying |

Note: K-Level is defined From Blooms Taxonomy

Contribution of Course Outcomes mapping with POs & PSOs (**1**- Low, **2** – Moderate,**3** – High)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **CO No.** | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** | **PSO1** | **PSO2** |
| CO1 | 3 | - | - | - | 3 | - | - | - | 3 | - | - | - | 3 | 3 |
| CO2 | 3 | - | - | - | 3 | - | - | - | 3 | - | - | - | 3 | 3 |

**COURSE CONTENT:**

1. Construct an AVL tree for a given set of elements which are stored in a file and implement insert and delete operation on the constructed tree. Write contents of treeinto a new file using in-order.
2. Construct B-Tree an order of 5 with a set of 100 random elements stored in array.Implement searching, insertion and deletion operations.
3. Construct Min and Max Heap using arrays, delete any element and display the content of the Heap.
4. Implement BFT and DFT for given graph, when graph is represented by

a) Adjacency Matrix b) Adjacency Lists

1. Write a program for finding the biconnected components in a given graph.
2. Implement Quick sort and Merge sort and observe the execution time for variousinput sizes (Average, Worst and Best cases).
3. Compare the performance of Single Source Shortest Paths using Greedy method when the graph is represented by adjacency matrix and adjacency lists.
4. Implement Job Sequencing with deadlines using Greedy strategy.
5. Write a program to solve 0/1 Knapsack problem Using Dynamic Programming.
6. Implement N-Queens Problem Using Backtracking.
7. Use Backtracking strategy to solve 0/1 Knapsack problem.
8. Implement Travelling Salesperson problem using Branch and Bound approach.

**TEXTBOOKS:**

1. Fundamentals of Data Structures in C++, HorowitzEllis, SahniSartaj, MehtaDinesh,

2ndEdition, Universities Press, 2008.

1. Computer Algorithms in C++, Ellis Horowitz, SartajSahni, SanguthevarRajasekaran,

2nd Edition, University Press, 2008.

**REFERENCE BOOKS:**

1. Data Structures and program design in C, Robert Kruse, 2nd Edition, Pearson Education Asia, 2006
2. An introduction to Data Structures with applications, Trembley& Sorenson, McGrawHill, 2nd edition, 2017

**E-RESOURCES/DIGITAL MATERIAL:**

1. <https://www.tutorialspoint.com/advanced_data_structures/index.asp>

<http://peterindia.net/Algorithms.html>

Regulation

**D23**

****

**ENVIRONMENTAL SCIENCE**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Course Category: | **Audit Course** | Credits: - | | | |
| Course Type: | **Theory** | Lecture-Tutorial-Practice: | 2 | 0 | 0 |
| Prerequisites | Science | Continuous Evaluation | 30M | | |
| Semester End Evaluation | - | | |
| Total Marks | - | | |

**Course Outcomes:**

|  |  |  |
| --- | --- | --- |
| After Successful Completion of course, the student will be able to: | | |
| CO No: | Course Outcome Description | K - Level |
| CO1 | Identify the multidisciplinary nature of environmental studies and various renewable and non-renewable resources. | Identifying |
| CO2 | Understand flow and bio-geo-chemical cycles and ecological pyramids. | Understanding |
| CO3 | Understand various causes of pollution and solid waste management and related preventive measures. | Understanding |
| CO4 | Understand rainwater harvesting, watershed management, ozone layer depletion and waste land reclamation | Understanding |
| CO5 | Predict population explosion and understand the value education, and welfare programs. | Predicting |

Note: K-Level is defined From Blooms Taxonomy

Contribution of Course Outcomes mapping with POs & PSOs (**1**- Low, **2** – Moderate,**3** – High)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **CO No.** | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** | **PSO1** | **PSO2** |
| CO1 | 3 | - | - | - | - | - | - | - | - | - | - | - | 3 | 3 |
| CO2 | 3 | - | - | - | - | - | - | - | - | - | - | - | 3 | 3 |
| CO3 | 3 | - | - | - | - | - | - | - | - | - | - | - | 3 | 3 |
| CO4 | - | 3 | - | - | - | - | - | - | - | - | - | - | 3 | 3 |
| CO5 | - | 3 | - | - | - | - | - | - | - | - | - | - | 3 | 3 |

**COURSE CONTENT:**

**UNIT–I:**

Multidisciplinary Nature Of Environmental Studies: Definition, Scope and Importance Need for Public Awareness. Natural Resources: Renewable and non-renewable resources – Natural resources and associated problems – Forest resources – Use and over – exploitation, deforestation, case studies – Timber extraction – Mining, dams and other effects on forest and tribal people – Water resources – Use and over utilization of surface and ground water – Floods, drought, conflicts over water, dams – benefits and problems–Mineral resources: Use and exploitation, environmental effects of extracting and using mineral resources, case studies–Food resources: World food problems, changes caused by agriculture and overgrazing, effects of modern agriculture, fertilizer-pesticide problems, water logging, salinity,case studies.–Energy resources:

# UNIT–II:

Ecosystems: Concep to fan ecosystem.–Structure and function of an ecosystem–Producers, consumers and decomposers – Energy flow in the ecosystem – Ecological succession – Food chains, food webs and ecological pyramids–Introduction, types, characteristic features, structure and function of the following ecosystem:

a. Forest ecosystem.

b. Grassl and ecosystem

c. Desert ecosystem

d. Aquatic ecosystems (ponds, streams, lakes, rivers, oceans, estuaries) Biodiversity

And Its Conservation: Introduction Definition: genetic, species and ecosystem diversity–Bio-geographical classification of India–Value of biodiversity: consumptive use, Productive use, social, ethical, aesthetic and option values – Biodiversity at global, National and local levels – India as a mega-diversity nation – Hot-sports of biodiversity – Threats to biodiversity: habitat loss, poaching of wildlife, man-wildlife conflicts– Endangered and endemic species of India –Conservation of biodiversity: In-situand Ex-situ conservation of biodiversity, bio-geo-chemical cycles

# UNIT–III:

Environmental Pollution: Definition, Cause, effects and control measures of:

a. Air Pollution.

b. Water pollution

c. Soil pollution

d. Marine pollution

e. Noise pollution

f. Thermal pollution

g. Nuclear hazards

Solid Waste Management: Causes, effects and control measures of urban and industrial wastes – Role of an individual in prevention of pollution – Pollution case studies – Disaster management: floods, earthquake, cyclone and landslides.

# UNIT–IV:

Social Issues and the Environment: From Unsustainable to Sustainable development– Urban problems related to energy – Water conservation, rain water harvesting, watershed management –Resettlement and rehabilitation of people; its problems and concerns. Case studies – Environmental ethics: Issues and possible solutions–Climate change, global warming, acid rain, ozone layer depletion, nuclear accidents and holocaust. Case Studies – Wastel and reclamation. – Consumerism and waste products. – Environment Protection Act. – Air (Prevention and Control of Pollution) Act. –Water (Prevention and control of Pollution) Act–Wild life Protection Act–Forest Conservation Act–Issues involved in enforcement of environment allegislation–Public awareness.

# UNIT–V:

Human Population And The Environment: Population growth, variation among nations. Population explosion – Family Welfare Programmes. – Environment and human health – Human Rights – Value Education–HIV/AIDS–Women and Child Welfare–Role of information Technology in Environment and human health–Case studies.

Field Work: Visit to a local area to document environmental assets River/ forest grassland/ hill/ mountain – Visit to a local polluted site-Urban/Rural/Industrial/Agricultural Study of common plants, insects, and birds–river, hills lopes, etc..

### Textbooks:

1. Text book of Environmental Studies for Undergraduate Courses Erach Bharucha for University Grants Commission, Universities Press..
2. Palani swamy, “Environmental Studies”, Pearson education .
3. S.AzeemUnnisa, “Environmental Studies” Academic Publishing Company
4. K.RaghavanNambiar,“Text book of Environmental Studies for Undergraduate Courses as per UGC model syllabus”, Scitech Publications (India), Pvt.Ltd

### Reference Books:

1. DeekshaDaveandE.SaiBabaReddy,“TextbookofEnvironmentalScience”,CengagePublications
2. M.AnjiReddy,“TextbookofEnvironmentalSciencesandTechnology”,BSPublication
3. J.P.Sharma, Comprehensive Environmental studies, Laxmi publications
4. J.GlynnHenryandGaryW.Heinke,“EnvironmentalSciencesandEngineering”,PrenticeHallofIndiaPrivatelimited
5. G.R.Chatwal,“ATextBookofEnvironmentalStudies”HimalayaPublishingHouse
6. GilbertM.MastersandWendellP.Ela,“IntroductiontoEnvironmentalEngineeringand Science, Prentice Hall of India Private limited

### E-Resources:

1. <https://onlinecourses.nptel.ac.in/noc23_hs155/preview>
2. <https://archive.nptel.ac.in/courses/120/108/120108004/>
3. <https://onlinecourses.nptel.ac.in/noc24_ge19/preview>
4. <https://onlinecourses.swayam2.ac.in/cec19_bt03/preview>
5. <https://www.classcentral.com/course/swayam-environmental-science-184135>
6. Abdul Bari,Introduction to Algorithms (youtube.com)
7. Introduction to the NPTEL MOOC on Design and Analysis of Algorithms by Prof. MUKUND MADHAVAN, IIT MADRAS,[Design and Analysis of Algorithms](Design%20and%20Analysis%20of%20Algorithms)<https://archive.nptel.ac.in/courses/106/106/106106131/>

**OBJECT ORIENTED PROGRAMMING THROUGH JAVA LAB**

****

Regulation

**D23**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Course Category: | PC | Credits : 3 | | | |
| Course Type: | Practical | Lecture-Tutorial-Practice: | - | - | 3 |
| Prerequisites | Good programming knowledge | Continuous Evaluation | 30-M | | |
| Semester End Evaluation | 70-M | | |
| Total Marks | 100--M | | |

**Course Outcomes:**

|  |  |  |
| --- | --- | --- |
| After Successful Completion of course, the student will be able to: | | |
| CO No: | Course Outcome Description | K - Level |
| CO1 | Apply the basic of concepts of programming in java. | Applying |
| CO2 | Apply the basic of concepts of Operations, Expressions, Control-flow, and Strings. | Applying |
| CO3 | Analyze different keywords in java | Analyzing |
| CO4 | Analyze the concepts of inheritance in java. | Analyzing |
| CO5 | Analyze applications using Exception Handling, Multi threading, Applet, Event Handling packages in java. | Analyzing |

Note: K-Level is defined From Blooms Taxonomy

Contribution of Course Outcomes mapping with POs & PSOs (**1**- Low, **2** – Moderate,**3** – High)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **CO No.** | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO1-** | **PO11** | **PO12** | **PSO1** | **PSO2** |
| CO1 | 3 | - | - | - | - | - | - | - | - | - | - | - | 3 | 3 |
| CO2 | 3 | - | - | - | - | - | - | - | - | - | - | - | 3 | 3 |
| CO3 | - | 3 | - | - | - | - | - | - | - | - | - | - | 3 | 3 |
| CO4 | - | 3 | - | - | - | - | - | - | - | - | - | - | 3 | 3 |
| CO5 | - | 3 | - | - | - | - | - | - | - | - | - | - | 3 | 3 |

**COURSE CONTENT:**

# Exercise - 1

# a) Write a JAVA program to display default value of all primitive data type of JAVA.

# b) Write a java program that display the roots of a quadratic equation ax2+bx=0. Calculate the discriminate D and basing on value of D, describe the nature of root.

# Exercise - 2

# a) Write a JAVA program to search for an element in a given list of elements using binary search

# Mechanism.

# b) Write a JAVA program to sort for an element in a given list of elements using bubble sort.

# c) Write a JAVA program using String Buffer to delete, remove character.

# Exercise - 3

# a) Write a JAVA program to implement class mechanism. Create a class, methods and invoke them inside main method.

# b) Write a JAVA program implement method overloading.

# c) Write a JAVA program to implement constructor.

# d ) Write a JAVA program to implement constructor overloading.

# Exercise - 4

# a) Write a JAVA program to implement Single Inheritance.

# b) Write a JAVA program to implement multi level Inheritance.

# c) Write a JAVA program for abstract class to find areas of different shapes

# Exercise - 5

# a) Write a JAVA program give example for “super” keyword.

# b) Write a JAVA program to implement Interface. What kind of Inheritance can be achieved?

# c) Write a JAVA program that implements Runtime polymorphism

# Exercise - 6

# a) Write a JAVA program that describes exception handling mechanism.

# b) Write a JAVA program Illustrating Multiple catch clauses.

# c) Write a JAVA program for creation of Java Built-in Exceptions.

# d) Write a JAVA program for creation of User Defined Exception.

# Exercise - 7

# a) Write a JAVA program that creates threads by extending Thread class. First thread display “Good Morning “every 1 sec, the second thread displays “Hello “every 2 seconds and the third display “Welcome” every 3 seconds,(Repeat the same by implementing Runnable)

# b) Write a program illustrating is Alive and join ()

# c) Write a Program illustrating Daemon Threads.

# d) Write a JAVA program Producer Consumer Problem

# Exercise – 8

# a) Write a JAVA program that import and use the user defined packages.

# b) Without writing any code, build a GUI that display text in label and image in an Image View (use Java FX)

# c) Build a Tip Calculator app using several Java FX components and learn how to respond to user

# Interactions with the GUI

# Exercise – 9

# a) Write a java program that connects to a database using JDBC

# b) Write a java program to connect to a database using JDBC and insert values into it.

# c) Write a java program to connect to a database using JDBC and delete values from it.

Course **PYTHON PROGRAMMING**

****

Regulation

**D23**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| CourseCategory: | SkillEnhancementcourse | Credits :2 | | | |
| Course Type: | Theory / **Tutorial** /**Practical** | Lecture-Tutorial-Practice: | 0 | 1 | 2 |
| Prerequisites | Good programming knowledge | Continuous Evaluation | 30M | | |
| Semester End Evaluation | 70M | | |
| Total Marks | 100M | | |

**Course Outcomes:**

|  |  |  |
| --- | --- | --- |
| After Successful Completion of course,the student will be able to: | | |
| CO No: | Course Outcome Description | K - Level |
| CO1 | **Apply**Python's fundamental components, **construct**control flow statements, **perform** input/output operations, **manage** exceptions, and **develop**well-structured, efficient, and robust Python programs. | Applying |
| CO2 | **Apply functions and their arguments, manage variable scope and lifetime, handle command line arguments, manipulate strings and lists using various operations and methods, and utilize built-in functions and commonly used modules to develop efficient Python programs.** | Applying |
| CO3 | **Applybuilt-in functions and methods on dictionaries, tuples, and sets, utilize tuples and sets for efficient data handling.** | Applying |
| CO4 | **Utilize file methods and modules to perform file operations, define and apply object-oriented programming principles by developing classes and objects, and implement encapsulation, inheritance, and polymorphism to construct robust and maintainable Python programs.** | Applying |
| CO5 | **Apply functional programming principles, work with JSON and XML data formats, utilize NumPy for numerical computations, and manipulate data efficiently using Pandas to perform data science tasks and analyses effectively.** | Applying |

Note: K-Level is defined From Blooms Taxonomy

Contribution of Course Outcomes mapping with POs & PSOs (**1**- Low, **2** – Moderate,**3** – High)

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **CO No.** | **PO1** | **PO2** | **PO3** | **PO4** | **PO5** | **PO6** | **PO7** | **PO8** | **PO9** | **PO10** | **PO11** | **PO12** | **PSO1** | **PSO2** |
| CO1 | 3 | - | - | - | 3 | - | - | - | - | - | - | - | 3 | 3 |
| CO2 | 3 | - | - | - | 3 | - | - | - | - | - | - | - | 3 | 3 |
| CO3 | 3 | - | - | - | 3 | - | - | - | - | - | - | - | 3 | 3 |
| CO4 | 3 | - | - | - | 3 | - | - | - | - | - | - | - | 3 | 3 |
| CO5 | 3 | - | - | - | 3 | - | - | - | - | - | - | - | 3 | 3 |

**COURSE CONTENT:**

UNIT-1:

History of Python Programming Language, Thrust Areas of Python, Installing AnacondaPython Distribution, Installing and Using Jupyter Notebook.

Parts of Python Programming Language: Identifiers, Keywords, Statements and Expressions,Variables, Operators, Precedence and Associativity, Data Types, Indentation, Comments,Reading Input, Print Output, Type Conversions, the type () Function and Is Operator,Dynamic and Strongly Typed Language.

Control Flow Statements: if statement, if-else statement, if...elif…else, Nested if statement,while Loop, for Loop, continue and break Statements, Catching Exceptions Using try andexcept Statement.

**Sample Experiments:**

1. Write a program to find the largest element among three Numbers.

2. Write a Program to display all prime numbers within an interval

3. Write a program to swap two numbers without using a temporary variable.

4.Demonstrate the following Operators in Python with suitable examples.

i) Arithmetic Operators ii) Relational Operators iii) Assignment Operatorsiv) LogicalOperators v) Bit wise Operators vi) Ternary Operator vii) Membership Operatorsviii)Identity Operators

5. Write a program to add and multiply complex numbers

6. Write a program to print multiplication table of a given number.

UNIT-2:

Functions: Built-In Functions, Commonly Used Modules, Function Definition and Callingthe function, return Statement and void Function, Scope and Lifetime of Variables, DefaultParameters, Keyword Arguments, \*args and \*\*kwargs, Command Line Arguments.

Strings: Creating and Storing Strings, Basic String Operations, Accessing Characters inString by Index Number, String Slicing and Joining, String Methods, Formatting Strings.

Lists: Creating Lists, Basic List Operations, Indexing and Slicing in Lists, Built-In FunctionsUsed on Lists, List Methods, del Statement.

**Sample Experiments:**

1. Write a program to define a function with multiple return values.

2. Write a program to define a function using default arguments.

3. Write a program to find the length of the string without using any library functions.

4. Write a program to check if the substring is present in a given string or not.

5. Write a program to perform the given operations on a list:

i. additionii. Insertioniii. slicing

6. Write a program to perform any 5 built-in functions by taking any list.

UNIT-3:

Dictionaries: Creating Dictionary, Accessing and Modifying key: value Pairs in Dictionaries,Built-In Functions Used on Dictionaries, Dictionary Methods, del Statement.

Tuples and Sets: Creating Tuples, Basic Tuple Operations, tuple () Function, Indexing andSlicing in Tuples, Built-In Functions Used on Tuples, Relation between Tuples and Lists,Relation between Tuples and Dictionaries, Usingzip () Function, Sets, Set Methods,Frozen set.

**Sample Experiments:**

1. Write a program to create tuples (name, age, address, college) for at least twomembers and concatenatethe tuples and print the concatenated tuples.
2. Write a program to count the number of vowels in a string (No control flow allowed)
3. Write a program to check if a given key exists in a dictionary or not.
4. Write a program to add a new key-value pair to an existing dictionary.
5. Write a program to sum all the items in a given dictionary.

UNIT-4:

Files: Types of Files, Creating and Reading Text Data, File Methods to Read and Write Data,Reading and Writing Binary Files, Pickle Module, Reading and Writing CSV Files, Pythonos and os.path Modules.

Object-Oriented Programming: Classes and Objects, Creating Classes in Python, CreatingObjects in Python, Constructor Method, Classes with Multiple Objects, Class Attributes VsData Attributes, Encapsulation, Inheritance, Polymorphism.

**Sample Experiments:**

1. Write a program to sort words in a file and put them in another file. The output fileshould have only lower-case words, so any upper-case words from source must belowered.
2. Python program to print each line of a file in reverse order.
3. Python program to compute the number of characters, words and lines in a file.
4. Write a program to create, display, append, insert and reverse the order of the itemsin the array.
5. Write a program to add, transpose and multiply two matrices.
6. Write a Python program to create a class that represents a shape. Include methods tocalculate its area and perimeter. Implement subclasses for different shapes like circle,triangle, and square.

UNIT-5:

Introduction to Data Science: Functional Programming, JSON and XML in Python, NumPywith Python, Pandas.

**Sample Experiments:**

1. Python program to check whether a JSON string contains complex object or not.
2. Python Program to demonstrate NumPy arrays creation using array () function.
3. Python program to demonstrate use of ndim, shape, size, dtype.
4. Python program to demonstrate basic slicing, integer and Boolean indexing.
5. Python program to find min, max, sum, cumulative sum of array
6. Create a dictionary with at least five keys and each key represent value as a list wherethis list contains at least ten values and convert this dictionary as a pandas data frameand explore the data through the data frame as follows:
   1. Apply head () function to the pandas data frame
   2. Perform various data selection operations on Data Frame
7. Select any two columns from the above data frame, and observe the change in oneattribute with respect to other attribute with scatter and plot operations in matplotlib

**TEXTBOOKS:**

1. Lambert, Kenneth A., and Martin Osborne. Fundamentals of PYTHON. Cengage, 2023.
2. Kurama, Vamsi. Python Programming: A Modern Approach. Pearson Education India, 2017.

**REFERENCE BOOKS:**

1. Gowrishankar, S., and A. Veena. Introduction to Python programming. Chapman and Hall/CRC, 2018.
2. Python Programming, S Sridhar, J Indumathi, V M Hariharan, 2ndEdition, Pearson,2024
3. Liang, Y. Daniel. Introduction to programming using Python. Pearson, 2013.

**E-RESOURCES/DIGITAL MATERIAL:**

1. https://www.coursera.org/learn/python-for-applied-data-science-ai
2. https://www.coursera.org/learn/python?specialization=python#syllabus