SCHEME OF INSTRUCTION AND SYLLABUS

B.Tech in Computer Science and Engineering (Artificial Intelligence and Machine Learning)

Regulation:VR23

w.e.f.2023-24

Department of Computer Science and Engineering

VELAGAPUDI RAMAKRISHNA SIDDHARTHA ENGINEERING COLLEGE

(An Autonomous, ISO 9001:2015 Certified Institution)
(Approved by AICTE, Accredited by NAAC with 'A+' Grade, Affiliated to JNTUK, Kakinada)
(Sponsored by Siddhartha Academy of General & Technical Education)
Vijayawada, Andhra Pradesh - 520007, INDIA.
www.vrsiddhartha.ac.in

Institute Vision

To nurture excellence in various fields of engineering by imparting timeless core values to the learners and to mould the institution into a centre of academic excellence and advanced research.

Institute Mission

To impart high quality technical education in order to mould the learners into globally competitive technocrats who are professionally deft, intellectually adept and socially responsible. The institution strives to make the learners inculcate and imbibe pragmatic perception and proactive nature so as to enable them to acquire a vision for exploration and an insight for advanced enquiry.

Department Vision

The department vision is clearly defined and is in line with the college's vision. The vision of the department is "To evolve as a centre of academic excellence and advanced research in Computer Science and Engineering discipline."

Department Mission

This mission of the department is concise and supports the college's mission. The mission of the Computer Science and Engineering department is "To inculcate students with profound understanding of fundamentals related to discipline, attitudes, skills, and their application in solving real world problems, with an inclination towards societal issues and research."

Program Educational Objectives
(Undergraduate)

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

Our program educational objectives are:

The graduates of the program will

1. Have knowledge and analytical skills, including mathematics, science and basic engineering.
2. Have in-depth learning skills to function productively as leadership role or as supportive members in multidisciplinary teams with effective communication.
3. Have extensive knowledge in state-of the- art frameworks in Artificial Intelligence to design industry accepted AI solutions using modern tools for allied domains with realistic constraints or pursue higher studies and continue to develop their professional knowledge.
4. Practice the profession with ethics, integrity, leadership and social responsibility.

Program Outcomes

On successful completion of the B.Tech in CSE (AI&ML) programme the student will be able to:

PO1 - *Engineering knowledge:* Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.

PO2 - *Problem analysis:* Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.

PO3 - *Design/development of solutions:* Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.

PO4 - *Conduct investigations of complex problems:* Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.

PO5 - *Modern tool usage:* Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.

PO6 - *The engineer and society:* Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.

PO7 - *Environment and sustainability:* Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.

PO8 - *Ethics:* Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.

PO9 - *Individual and team work:* Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.

PO10 - *Communication:* Communicate effectively on complex engineering activities with the engineering community and with society at large, such as being able to comprehend and write effective reports and design documentation, make effective presentations, and give and receive clear instructions.

PO11 - *Project management and finance:* Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.

PO12 - *Lifelong learning:* Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

Program Specific Outcomes

PSO1: Develop software applications or solutions as per the needs of Industry and society.

PSO2: Adopt new and fast emerging technologies in computer science and engineering.

COURSE CATEGORY ABBREVIATIONS

1. Humanities and Sciences-HS
2. Basic Sciences-BS
3. Engineering Science-ES
4. Program Core-PC
5. Soft Skills-SS
6. Skill Enhancement Course-SEC
7. Audit Course-AC
8. Mandatory Course-MC
9. Program Elective-PE
10. Open Elective-OE
11. Humanities and Social Sciences-HSS
12. Advanced Skill Course

SCHEME OF INSTRUCTION

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

(ARTIFICIAL INTELLIGENCE & MACHINE LEARNING)

SCHEME OF INSTRUCTION FOR FOUR YEAR UG PROGRAMME [VR23]

Semester III

				Contact Hours			25
S. No.	Course Code	Course Category	Course Name	L	T	P	C
1	23ES3101B	ES	Artificial Intelligence	3	0	0	3
2	23HS3102	BSH	Universal Human Values 2 -Understanding Harmony	2	1	0	3
3	23ES3103B	ES	Discrete Mathematics	3	0	0	3
4	23AI&ML3304	PC	Advanced Data Structures & Algorithms Analysis	3	0	0	3
5	23AI&ML3305	PC	Object Oriented Programming Through Java	3	0	0	3
6	23TP3106	SS-1	Logic & Reasoning	0	0	2	1
7	23AI&ML3651	SEC	Python Programming Lab	0	0	2	1
8	23AI&ML3352	PC Lab 1	Advanced Data Structures and Algorithms Lab	0	0	3	1.5
9	23AI&ML3353	PC Lab 2	Object Oriented Programming Through Java Lab	0	0	3	1.5
			Total	14	1	10	20

[*L - Lecture, T - Tutorial, P - Practical, C - Credits*]

B.Tech in CSE (Artificial Intelligence and Machine Learning)

Scheme of Instruction - VR23

Semester IV

						Contact Hours	29
S. No.	Course Code	Course Category	Course Name	L	T	P	C
1	23HS4101C	ES	Optimization Techniques	2	0	0	2
2	23BS4102B	BS	Probability & Statistics	3	0	0	3
3	23AI&ML4303	PC	Machine Learning	3	0	0	3
4	23AI&ML4304	PC	Database Management Systems	3	0	0	3
5	23AI&ML4305	PC	Digital Logic & Computer Organization	3	0	0	3
6	23TP4106	SS-2	English for Professionals	0	0	2	1
7	23MC4107	AC	Environmental Science	2	0	0	-
8	23AI&ML4651	SEC	Industry Coding Practice-I	0	0	2	1
9	23ES4152	ES	Design Thinking & Innovation	1	0	2	2
10	23AI&ML4353	PC Lab-1	Machine Learning Lab	0	0	3	1.5
11	23AI&ML4354	PC Lab-2	Database Management Systems Lab	0	0	3	1.5
			Total	17	0	12	21
Summer Internship 6 weeks (Mandatory) during summer vacation (EPICS)							
Honors/ Minor Courses (the hours distribution can be 4-0-0, 3-0-2 or 3-1-0 also)				4	0	0	4

[*L - Lecture, T - Tutorial, P - Practical, C - Credits*]

B.Tech in CSE (Artificial Intelligence and Machine Learning)

Scheme of Instruction - VR23

Semester V

						Contact Hours	27
S. No.	Course Code	Course Category	Course Name	L	T	P	C
1	23AI&ML5301	PC	Information Retrieval Systems	3	0	0	3
2	23AI&ML5302	PC	Computer Networks	3	0	0	3
3	23AI&ML5303	PC	Automata Theory & Compiler Design	3	0	0	3
4	23AI&ML5404	PE I	1. Software Engineering 2. Cloud Computing 3. Internet of Things 4. Exploratory Data Analysis with Python	3	0	0	3
5	23AI&ML5205	OE-I / Minor-1		3	0	0	3
6	23AI&ML5351	PC Lab - 1	Information Retrieval Lab	0	0	3	1.5
7	23AI&ML5352	PC Lab - 2	Computer Networks Lab	0	0	3	1.5
8		SEC	Industry Coding Practice-II	0	0	2	1
9		SS - 3	Personality Development	0	0	2	1
10		HSS	Advanced Communication skills lab	0	0	2	1
11		Evaluation of Community Service Internship	EPICS	-	-	-	2
			Total	15	0	12	23
Honors/ Minor Courses (the hours distribution can be 4-0-0, 3-0-2 or 3-1-0 also)				4	0	0	4

[*L - Lecture, T - Tutorial, P - Practical, C - Credits*]

B.Tech in CSE (Artificial Intelligence and Machine Learning)

Scheme of Instruction - VR23

Semester VI

				Contact Hours			30
S. No	Course Code	Course Category	Course Name	L	T	P	C
1	23AI&ML6301	PC	Natural Language Processing	3	0	0	3
2	23AI&ML6302	PC	Deep Learning	3	0	0	3
3	23AI&ML6303	PC	Data Visualization	2	0	0	2
4	23AI&ML6404	PE II	1. Software Testing Methodology 2. Cryptography & Network Security 3. DevOps 4. Recommender Systems 5. Any of the 12-Week SWAYAM / NPTEL Course suggested by the BoS	3	0	0	3
5	23AI&ML6405	PE III	1. Software Project Management 2. Mobile Adhoc Networks 3. Computer Vision 4. Soft computing 5. Any of the 12-Week SWAYAM /NPTEL Course suggested by the BoS	3	0	0	3
6	23AI&ML6206	OE 2		3	0	0	3
7	23AI&ML6351	PC Lab-1	Deep Learning Lab	0	0	4	2
8	23AI&ML6352	PC Lab-2	Data Visualization Lab(Using Power BI / Tableau etc.,)	0	0	4	2
9		SS - 4	Quantitative Aptitude	0	0	2	1
10		AC	Technical Paper Writing & IPR	2	0	0	-
			Total	20	0	10	23
Honors/ Minor Courses (the hours distribution can be 4-0-0, 3-0-2 or 3-1-0 also)				4	0	0	4

[*L - Lecture, T - Tutorial, P - Practical, C - Credits*]

B.Tech in CSE (Artificial Intelligence and Machine Learning)

Semester VII

			Contact Hours				21
S. No.	Course Code	Course Category	Course Name	L	T	P	C
1	23AI&ML7301	PC	Augmented Reality and Virtual Reality	3	0	0	3
2	23AI&ML7402	HSS	Human Resource Management	2	0	0	2
3	23AI&ML7403	PE IV	1. Responsible AI (Ethics, privacy, copyright, data governance) 2. Blockchain Technology 3. NoSQL databases 4. Quantum Computing 5. Any of the 12-Week SWAYAM /NPTEL Course suggested by the BoS	3	0	0	3
4	20AI&ML7404	PE V	1. Agile methodologies 2. Big Data Analytics 3. Reinforcement Learning 4. High Performance Computing 5.Any of the 12-Week SWAYAM /NPTEL Course suggested by the BoS	3	0	0	3
5	20AI&ML7205	OE III	OE III	3	0	0	3
6	20AI&ML7206	OE IV	OE IV	3	0	0	3
7		ASC	Corporate Readiness Skills	0	0	2	2
8		AC	Constitution of India	2	0	0	-
9		Internship	Evaluation of Industry Internship	-	-	-	2
			Total	19	0	2	21
Industrial/ Research Internship six weeks (Mandatory) during summer vacation							
Honors/ Minor Courses (the hours distribution can be 4-0-0, 3-0-2 or 3-1-0 also)				0	0	0	4

L - Lecture, T - Tutorial, P - Practical, C - Credits]

Note: Open Elective III and Open Elective IV are self-learning. Students may opt from any MOOCs platforms. They have to submit the certificate before the last instruction day of VII semester. Course selection for MOOCs is subject to approval by the Head of the Department.

B.Tech in CSE (Artificial Intelligence and Machine Learning)

Scheme of Instruction - VR23

Semester VIII

						Contact Hours	24
S. No.	Course Code	Course Category	Course Name	L	T	P	C
1	20AI&ML8551	Internship/ Project**	Major Project and Internship (6 Months)	0	0	24	12
			Total	0	0	24	12

[*L - Lecture, T - Tutorial, P - Practical, C - Credits*]

Note: The student should undergo internship and simultaneously heshe should work on a project with well-defined objectives. At the end of the semester the student should submit an internship completion certificate and a project report. If any of our associated company comes forward to offer an emerging course that will be offered as an industry offered course in V, VI or VII semesters under program elective with the approval of BoS. This is incorporated to enhance student skills and employability in cutting edge technologies.

Minor: To obtain Minor Engineering, student needs to obtain 18 credits by successfully completing any of the following courses in the concern stream.

For Minor in AI & ML:

1. Advanced Data Structures & Algorithm Analysis	3-0-3
2. Artificial Intelligence	3-0-3
3. Operating Systems	3-0-3
4. Computer Networks	3-0-0
5. Machine Learning	3-0-0
6. Deep Learning	3-0-0
7. Cloud computing	

For Minor in Cyber Security:

1. Operating Systems	3-0-3
2. Computer Networks	3-0-0
3. Artificial Intelligence	3-0-3
4. Cloud computing	3-0-0
5. Cyber Security	3-0-0
6. Cryptography & Network Security	3-0-3
7. Blockchain technology	3-0-0

For Minor in Data Science:

1. Advanced Data Structures & Algorithm Analysis	3-0-0
2. Artificial Intelligence	3-0-0
3. Introduction to Data Science	3-0-3
4. Machine Learning	3-0-0
5. Data Engineering	3-0-3
6. Big Data Analytics	3-0-0
7. Cloud computing	3-0-0

Open Electives, offered to other department students:

Open Elective I: Java Programming

Open Elective II: Operating Systems

Open Elective III: Data Base Management Systems

Open Elective IV: Computer Networks

Honors: Student need to obtain 18 Credits by successfully completing any of the following

1. Machine Learning for Cyber Security	3-0-0
2. Robotics	3-0-0
3. Explainable AI	3-0-0
4. Blockchain and its Applications – NPTEL 12W	3-0-0
5. Applied Linear Algebra in AI & ML (MOOCS- SWAYAM / NPTEL 12W)	3-0-0
6. Applied Time-Series Analysis (MOOCS- SWAYAM / NPTEL 12W)	3-0-0
7. Machine Learning for Engineering and Science Applications (MOOCS- SWAYAM / NPTEL 12W)	3-0-0

8. Parallel Computer Architecture (MOOCS- SWAYAM / NPTEL 12W) 3-0-0
9. Principles of Knowledge Graphs 3-0-0
10. Any of the SWAYAM /NPTEL Course suggested by the BoS Chairman

SEMESTER III

23ES3101B
ARTIFICIAL INTELLIGENCE

Course Category	Engineering Science	Credits	3
Course Type	Theory	L-T-P	3-0-0
Prerequisites	23BS1101 Linear Algebra & Calculus	Continuous Eval Semester End Eval Total Marks	30 70 100

COURSE OUTCOMES

1. Understand the AI agents for problem solving.
2. Understand AI solutions using Heuristic and Pruning approaches.
3. Apply various knowledge representation techniques.
4. Apply the logic concepts and learnings to Expert Systems.

Contribution of Course Outcomes towards achievement of Program Outcomes (1 – Low, 2 - Medium, 3 – High)

CO	PO												PSO		BTL	POI
	1	2	3	4	5	6	7	8	9	10	11	12	1	2		
1	2	2											1	1	2	1.2.1, 1.6.1, 1.7.1, 2.5.2, 2.6.2
2	1	2	2										1	1	2	1.6.1, 1.7.1, 2.5.2, 2.6.4, 2.8.1, 3.5.1
3	1	2	2										1	1	3	1.6.1, 1.7.1, 2.5.2, 2.6.4, 3.5.1
4	2	2	2										1	1	3	1.6.1, 1.7.1, 2.5.2, 2.6.4, 2.7.1, 3.5.1

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:Well defined problems and Solutions, problem formulation, Example problems- 8-puzzle problem, Water jug problem.

UNIT II

Searching: Searching for solutions, uniformed search strategies – Breadth first search, depth first Search.

Heuristic Search: Search with partial information, Informed search strategies, Hill climbing, Best First search, A* Algorithm, Problem Reduction -AO* Algorithms..

Adversarial Search: Games, mini-max algorithm, Problem in Game playing, Alpha-Beta pruning, Evaluation functions.

UNIT III

Representation of Knowledge: Weak Slot and filler structures, Semantic nets-frames and frame systems.
Representing knowledge using rules: Logic programming, Rules based deduction systems.
Knowledge Representation Issues: Representation and mapping, Approaches in knowledge representation, Issues in knowledge representation.

UNIT IV

Logic concepts: Predicate Logic, Propositional vs. Predicate Logic, unification & lifts forward chaining, Backward chaining, Resolution.
Learning: Learning from observation, Explanation based learning, Statistical Learning methods, Reinforcement Learning.
Expert Systems: Architecture of expert systems, Roles of expert systems, Knowledge Acquisition, Typical expert systems – MYCIN, DART, Expert systems shells.

TEXTBOOKS

1. S. Russel and P. Norvig, Artificial Intelligence – A Modern Approach, Pearson Education, 2^{nd} Edition, 2007.
2. Kevin Night and Elaine Rich, Nair B, Artificial Intelligence (SIE), Mc Graw Hill, 3^{rd} Edition, 2010.

REFERENCE BOOKS

1. David Poole, Alan Mackworth, Randy Goebel, Computational Intelligence: a Logical approach, Oxford University Press, 1^{st} Edition, 2004.
2. G. Luger, Artificial Intelligence: Structures and Strategies for complex problem-solving, Pearson Education, 4^{th} Edition, 2001
3. J. Nilsson, Artificial Intelligence: A new Synthesis, Elsevier Publishers, 1^{st} Edition, 2003.
4. Saroj Kaushik, Artificial Intelligence, Cengage Learning, 1^{st} Edition, 2011.

E-RESOURCES AND OTHER DIGITAL MATERIALS

1. Fundamentals of Artificial Intelligence, https://swayam.gov.in/\nd1_noc19_me71/preview, Last Accessed On: 31-05-2024.

23HS3102

UNIVERSAL HUMAN VALUES 2: UNDERSTANDING HARMONY

Course Category	Basic Sciences and Humanities	Credits	3
Course Type	Mandatory course (suggested by AICTE)	L-T-P	2-1-0
Prerequisites	Universal Human Values 1	Continuous Eval Semester End Eval Total Marks	30 70 100

COURSE OUTCOMES

1. Understand and aware of themselves and their surroundings(family, society and nature).
2. Handle problems with sustainable solutions, while keeping human relationships and human nature in mind.
3. Exhibit critical ability and become sensitive to their commitment towards their understanding of human values, human relationship and human society.
4. Apply what they have learnt to their own self in different day-to-day settings in real life.

Contribution of Course Outcomes towards achievement of Program Outcomes (1 – Low, 2 - Medium, 3 – High)

CO	PO												PSO		BTL	POI
	1	2	3	4	5	6	7	8	9	10	11	12	1	2		
1						1			2						2	6.1.1, 9.2.1, 9.2.2, 9.2.3
2			3												2	3.1.1, 3.2.3
3						2									2	6.2.1
4								3				2			3	8.1.1, 8.2.2 12.1.1, 12.2.2

COURSE CONTENT

UNIT I

Course introduction, need, basic guidelines, content and process for value education:

Part-1: Purpose and motivation for the course, recapitulation from UHV-I, Self-exploration: what is it?, its content and process, 'Natural acceptance' and experiential validation- as the process for self-exploration. Continuous Happiness and Prosperity

– A look at basic Human Aspirations.

Part-2: Right understanding, Relationship and Physical Facility – the basic requirements for fulfillment of aspirations of every human being with their correct priority, Understanding Happiness and Prosperity correctly – A critical appraisal of the current scenario, Method to fulfill the above human aspirations: understanding and living in harmony at various levels. (Practice sessions are to be included to discuss natural acceptance in human being as the innate acceptance for living with responsibility (living in relationship, harmony and co-existence) rather than as arbitrariness in choice based on liking-disliking).

UNIT II ***Understanding Harmony in the Human Being – Harmony in Myself:***

Part-1: Understanding human being as a co-existence of the sentient 'I' and the material 'Body'. Understanding the needs of Self ('I') and 'Body' – happiness and physical facility, Understanding the Body as an instrument of 'I' (I being the doer, seer and enjoyer).

Part-2: Understanding the characteristics and activities of 'I' and harmony in 'I'. Understanding the harmony of I with the Body: Sanyam and Health; correct appraisal of Physical needs, meaning of Prosperity in detail, Programs to ensure Sanyam and Health. (Practice sessions are to be included to discuss the role others have played in making material goods available to me. Identifying from one's own life. Differentiate between prosperity and accumulation. Discuss program for ensuring health vs. dealing with disease).

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

Part-1: Understanding values in human-human relationship; meaning of Justice (nine universal values in relationships) and program for its fulfillment to ensure mutual happiness; Trust and Respect as the foundational values of relationship, Understanding the meaning of Trust; Difference between intention and competence, Understanding the meaning of Respect, Difference between respect and differentiation; the other salient values in relationship.

Part-2: Understanding the harmony in the society (society being an extension of family); Resolution, Prosperity, fearlessness (trust) and co-existence as comprehensive Human Goals, Visualizing a universal harmonious order in society–Undivided Society, Universal Order–from family to world family. (Practice sessions are to be included to reflect on relationships in family, hostel and institute as extended family, real life examples, teacher-student relationship, goal of education, etc. Gratitude as a universal value in relationships. Discuss with scenarios. Elicit examples from students' lives).

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

Part-1: Understanding the harmony in the Nature, Inter-connectedness and mutual fulfillment among the four orders of Nature – recyclability and self-regulation in nature, Understanding Existence as Co-existence of mutually interacting units in all-pervasive space, Holistic perception of harmony at all levels of existence.

Part-2: Natural acceptance of human values, Definitiveness of ethical human conduct, Basis for humanistic education, humanistic constitution and humanistic universal order, Competence in professional ethics: a) ability to utilize the professional competence for augmenting universal human order, b) ability to identify the scope and characteristics of people-friendly and eco-friendly production systems, c) ability to identify and develop appropriate technologies and management patterns for above production systems, Case studies of typical holistic technologies, management models and production systems, Strategy for transition from the present state to Universal Human Order: a) at the level of individual: as socially and ecologically responsible engineers, technologists and managers, b) at the level of society: as mutually enriching institutions and organizations.

(Part-1:Practice sessions are to be included to discuss human being as cause of imbalance in nature (film "Home" can be used), pollution, depletion of resources and role of technology, etc.

Part-2: Practice exercises and case studies are to be taken up in practice (tutorial) sessions eg. to discuss the conduct as an engineer or scientist, etc.)

TEXTBOOKS

1. R. R. Gaur, R. Asthana and G. P. Bagaria, A Foundation Course in Human Values and Professional Ethics, Excel Books Private Limited, New Delhi, 2^{nd} Revised Edition, 2019.
2. R. R. Gaur, R. Sangal and G. P. Bagaria, A Foundation Course in Human Values and Professional Ethics, Excel Books Private Limited, New Delhi, 1^{st} Edition, 2010

REFERENCE BOOKS

1. Pandit Sunderlal, Prabhath Prakashan, Bharat Mein Angreji Raj, Delhi , 2018
2. J. C. Kumarappa, Sarva-Seva-Sangh Prakashan, Economy of Permanence, Varanasi, 2017.
3. Annie Leonard, The Story of Stuff: The impact of overconsumption on the planet, our communities, and our health and how we can make it better, Free Press, New York, 2010.
4. Romain Rolland, Advaitha Ashrama, The Life of Vivekananda and the Universal gospel, India, 2010.
5. Mohandas Karamchand Gandhi, The story of my experiments with truth: Mahatma Gandhi Autobiography, B. N. Publishing, 2008.
6. Cecile Andrews, Slow is beautiful: New Visions of Community, New Society Pub-

lishers, Canada, 2006.
7. A. N. Tripathi, Human Values, New Age International Publishers, New Delhi, 2004.
8. Dharampal, Rediscovering India, Society for Integrated Development of Himilaya, 2003.
9. Romain Rolland, Srishti Publishers & Distributors, Mahatma Gandhi: The Man who become one with the Universal Being, New Delhi, 2002.
10. A. Nagaraj, JeevanVidya Prakashan, Amarkantak, JeevanVidya: EkParichaya, 1999.
11. Maulana Abul Kalam Azad, Orient Blackswan, India Wins Freedom: The Complete Version, 1988.
12. E. F. Schumacher, Small is beautiful: A study of economics as if people mattered, Vintage Books, London, 1993

E-RESOURCES AND OTHER DIGITAL MATERIALS

1. A Foundation Course in Human Values and Professional Ethics, https://dokumen.pub/a-foundation-course-in-human-values-and-professional-ethicsfirstnbsped-9788174467812.html, Last Accessed On: 12/05/2024.
2. AICTE – SIP, https://www.youtube.com/channel/UCo8MpJB_aaVwB4LWLAx6AhQ, Last Accessed On: 16/3/2024.
3. AICTE UHV Teaching Learning Material, https://fdp-si.aicte-india.org/download.php#1, Last Accessed On: 28/3/2024.

23ES3103B
DISCRETE MATHEMATICS

Course Category	Engineering Science	Credits	3
Course Type	Theory	L-T-P	3-0-0
Prerequisites	-	Continuous Eval Semester End Eval Total Marks	30 70 100

COURSE OUTCOMES

1. Analyze of propositional calculus and first order logic.
2. Apply the basic and advanced counting techniques.
3. Analyze of relations and digraphs and their applications.
4. Analyze of graphs and their applications.

Contribution of Course Outcomes towards achievement of Program Outcomes (1 – Low, 2 - Medium, 3 – High)

CO	PO												PSO		BTL	POI
	1	2	3	4	5	6	7	8	9	10	11	12	1	2		
1	3	3		1								1			4	1.2.1, 1.7.1, 2.5.1, 4.6.3, 12.4.2
2	3	3		1								1			3	1.2.1, 2.5.1, 2.5.3, 4.6.3, 12.4.2
3	3	3		1								1			4	1.2.1, 2.8.1, 4.4.1, 12.4.2
4	3	3		1								1			4	1.2.1, 2.5.3, 4.4.1, 12.4.2

COURSE CONTENT

UNIT I: Propositional Calculus

Fundamentals of Logic:

Propositions, Connectives, Propositional functions, Truth tables, Tautology, Contradiction, Logical equivalences, Normal forms, Logical inferences, Methods of proof of an implication.

First Order Predicate Logic: Predicate, Quantifiers, Rules of inference for Quantified propositions.

UNIT II: Counting Techniques

Basics of Counting:

Sum and product rules, Indirect counting, One to one correspondence, Combinations and permutations, Enumerating combinations and permutations with and without

repetitions.
Advanced Counting Techniques: Generating function of sequences, Recurrence relations, Solving recurrence relations – substitution- Generating functions-The method of characteristic roots, Solution of inhomogeneous recurrences relations.

UNIT III: Relations and Digraphs
Relations and basic graphs, Special properties of binary relations, Equivalence relation, Partially ordered sets, Hasse diagrams, Lattices, Operations on relations, Paths and closures, Directed graphs and Adjacency matrices, Transitive closure, Warshall's algorithm.

UNIT IV: Graph Theory
Introduction(graphs, sub graphs, circuits, trees) Sum of degrees theorem, Isomorphism and sub graphs, planar graphs, Euler's formula, Multi graphs and Euler's circuits, Hamiltonian graphs, Grin-berg's theorem, Graph coloring, Chromatic numbers.

TEXTBOOKS

1. Joe L. Mott, Abraham Kandel, and Theodore P. Baker, Discrete Mathematics for Computer Scientists & Mathematicians, PHI, 2^{nd}Edition, 2008.

REFERENCE BOOKS

1. J. P. Tremblay and R.Manohar, Discrete Mathematical structures with applications to computer science, MGHE, 1^{st}Edition, 2017.
2. K. H. Rosen, Discrete Mathematics and its Applications, Mc Graw Hill Companies, 7^{th}Edition, 2011.
3. D. S. Malik and M. K. Sen, Discrete Mathematical Structures: Theory and applications, Revised Edition, 2012.

E-RESOURCES AND OTHER DIGITAL MATERIALS

1. Prof. Arindama Singh Department of Maths IIT Madras, https://nptel.ac.in/courses/111/106/111106052/, Last accessed On: 25/03/2024.
2. Prof Soumen Maity, Department of Maths, IISER Pune,https://nptel.ac.in/courses/111/106/111106102/, Last accessed On: 15/04/2024.
3. Dr.L.Sunil Chandran, Department of Computer Science and Automation, IISC Bangalore, https://nptel.ac.in/courses/106/108/106108051/, Last accessed On: 15/03/2024.

23AI&ML3304

ADVANCED DATA STRUCTURES & ALGORITHMS ANALYSIS

Course Category	Program Core	Credits	3
Course Type	Theory	L-T-P	3-0-0
Prerequisites	23PC2104A Data Structures	Continuous Eval Semester End Eval Total Marks	30 70 100

COURSE OUTCOMES

1. Understand the functions of different nonlinear data structures- binary trees, search trees.
2. Apply graph-based algorithms, design technique - divide and conquer, and greedy methods for solving complex problems.
3. Apply dynamic programming and backtracking design techniques to solve complex problems
4. Apply the concepts of Branch and Bound techniques to solve complex problems; Understand NP-Hard, and NP-Complete problems.

Contribution of Course Outcomes towards achievement of Program Outcomes (1 – Low, 2 - Medium, 3 – High)

CO	PO												PSO		BTL	POI
	1	2	3	4	5	6	7	8	9	10	11	12	1	2		
1	1	2											1	1	2	1.2.1, 1.6.1, 1.7.1, 2.5.2
2	1	2	3										1	1	3	1.2.1, 1.6.1, 1.7.1, 2.5.2, 2.6.4, 3.5.1
3	1	2	3										1	1	3	1.2.1, 1.6.1, 1.7.1, 2.5.2, 2.7.1, 3.5.1
4	1	2	3										1	1	3	1.2.1, 1.6.1, 1.7.1, 2.5.2, 2.6.4, 2.7.1, 3.5.1

COURSE CONTENT

UNIT I

Introduction to Algorithm Analysis: Space and Time Complexity analysis, Asymptotic Notations.

AVL Trees Creation, Insertion, Deletion operations and Applications.

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

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

UNIT II

Graphs: Terminology, Representations, Basic Search and Traversals, Connected Components and Biconnected Components, applications
Divide and Conquer: The General Method, Quick Sort, Merge Sort, Stassen's matrix multiplication, Convex Hull
Greedy Method: General Method, Job Sequencing with deadlines, Knapsack Problem, Minimum cost spanning trees, Single Source Shortest Paths.

UNIT III

Dynamic Programming: General Method, All pairs shortest paths, Single Source Shortest Paths– General Weights (Bellman Ford Algorithm), Optimal Binary Search Trees, 0/1 Knapsack, String Editing, Travelling Salesperson problem.
Backtracking: General Method, 8-Queens Problem, Sum of Subsets problem, Graph Coloring, 0/1 Knapsack Problem.

UNIT IV

Branch and Bound:The General Method,0/1Knapsack Problem, Travelling Salesperson problem.
NP Hard and NP Complete Problems: Basic Concepts, Cook's theorem.
NP Hard Graph Problems:Clique Decision Problem (CDP), Chromatic Number Decision Problem (CNDP), Traveling Salesperson Decision Problem (TSP).
NP Hard Scheduling Problems:Scheduling Identical Processors, Job Shop Scheduling.

TEXTBOOKS

1. Reema Thareja, Data Structures using C, Oxford University Press, 2^{nd} Edition, 2014. (Unit I)
2. Ellis Horowitz, Satraj Sahni and Rajasekharan, Fundamentals of Computer Algorithms, Galgotia Publications Pvt. Ltd, 2^{nd} Edition, 2008. (Unit II, III, and IV).

REFERENCE BOOKS

1. Horowitz Sahni and Anderson-Freed, Fundamentals of Data Structures in C, Universities Press, 2^{nd} Edition, 2008.
2. Richard F.Gilberg & B.A.Forouzan, Data Structures A Pseudo code Approach with C, Cengage Learning, 2^{nd} Edition, 2007.
3. M.T.Goodrich and R.Tomassia, Algorithm Design: Foundations, Analysis and Internet examples, John Wiley and sons, 2^{nd} Edition, 2006.
4. T.H.Cormen, C.E.Leiserson, R.L.Rivest and C.Stein, Introduction to Algorithms, Mit Press, 3^{rd} Edition, 2009.
5. Allen Weiss, Data structures and Algorithm Analysis in C, Pearson Education,

2^{nd} Edition, 2002.

E-RESOURCES AND OTHER DIGITAL MATERIALS

1. Advanced Data Structures Course, https://www.tutorialspoint.com/advanceddatastructures/index.asp, Last Accessed On: 31/5/2024.
2. Computer Algorithms, http://peterindia.net/Algorithms.html, Last Accessed on: 24-06-2024.

23AI&ML3305
OBJECT ORIENTED PROGRAMMING THROUGH JAVA

Course Category	Program Core	Credits	3
Course Type	Theory	L-T-P	3-0-0
Prerequisites	23ES1104 Introduction to Programming	Continuous Eval Semester End Eval Total Marks	30 70 100

COURSE OUTCOMES

1. Understand the basic concepts of object oriented programming.
2. Interpret multiple inheritances through interfaces for a given application.
3. Apply exceptions, thread capabilities and handling files on a given application.
4. Illustrate the use of functional programming and Collections framework for a given application.

Contribution of Course Outcomes towards achievement of Program Outcomes (1 – Low, 2 - Medium, 3 – High)

CO	PO												PSO		BTL	POI
	1	2	3	4	5	6	7	8	9	10	11	12	1	2		
1	2	3													2	1.7.1, 2.5.1, 2.5.2, 2.7.1
2	2	2	3										2	2	3	1.7.1, 2.5.1, 2.5.2, 2.6.3, 2.7.1, 3.5.1, 3.8.2
3	2	2	3										2	2	3	1.7.1, 2.5.1, 2.5.2, 2.6.3, 2.6.4, 3.5.1, 3.8.2
4	1	2	3	2	2								2	2	3	1.7.1, 2.5.1, 2.5.2, 2.6.3, 2.6.4, 3.5.1, 3.6.2, 3.8.2, 4.6.2, 5.4.2

COURSE CONTENT

UNIT I

Introduction, The History and Evolution of Java: Java history and evolution, Java features, Java's Magic: Byte Code, How Java differs from C and C++.
An Overview of Java: Object Oriented Programming: Two paradigms, Principles of OOP, A First simple Program and Control statements. Data Types, Variables and Arrays: Java keywords, Primitive types, Integers, Floating-Point Types, Characters, Booleans, Variables, Operators, Type Conversion, Casting and Arrays.

Introducing Classes and Objects:Class fundamentals, declaring objects, assigning object reference variables, introducing methods, constructors, this keyword, Garbage collection, overloading methods, using objects as parameters, returning objects, static and final keywords, nested and inner classes.

UNIT II

String Handling: The String Constructors, String Buffer Class, String Tokenizer class.
Inheritance: Inheritance basics, using super, multilevel hierarchy, method overriding, dynamic method dispatch, using abstract classes, final with inheritance.
Packages & Interfaces: Defining a package, finding package and CLASSPATH. Access protection, importing packages, Defining an interface, implementing interfaces, nested interfaces, applying interfaces, variables in interfaces.

UNIT III

Exception handling: Exception handling fundamentals, exception types, uncaught exceptions, using try and catch, multiple catch clauses, throw, throws, finally, creating your own exception subclasses.
I/O streams: Byte Streams- Input Stream, Output Stream, File Input Stream, File Output Stream, Character Streams- Reader, Writer, File Reader, and File Writer.
Multithread Programming: The Java Thread Model, Creating a thread: Implementing Runnable, Extending Thread, creating multiple threads, Thread Priorities, Synchronization- Using Synchronized methods, The synchronized Statement.

UNIT IV

Pragmatic Functional Programming using Lambdas: Introduction to Functional programming, Functional Programming concepts and terminology, Functional Interfaces, Working with Lambda Expressions and Method References.
Collections Framework: Collections overview, Collection interfaces: Collection, List, and Set. Collection Classes: Array List, Linked List, Hash Set. Map Classes: Hash Map, Tree Map
The Stream API: Stream basics, Reduction operations, Using parallel streams, Mapping, Collecting, Iterators and Streams.

TEXTBOOKS

1. Herbert Schildt, Java The Complete Reference, Oracle Press, 11^{th} Edition, 2019.

REFERENCE BOOKS

1. Herbert Schildt, Dale Skrien, Java Fundamentals: A Comprehension Introduction, McGraw-Hill Education India Pvt., Special Indian Edition, Ltd, 2017.
2. E Balaguruswamy, Programming with Java, Mc Graw Hill, 4^{th} Edition, 2020.
3. Paul J. Dietel and Dr.Harvey M. Deitel, "Java How to Program", Deitel & Associates, 11^{th} Edition, 2018.

4. Timothy Budd, Understanding Object Oriented Programming with Java, Pearson Education, Updated edition, 2013.
5. Kathy Sierra & Bert Bates, Head First Java, Oreilly, 2^{nd} Edition, 2023.

E-RESOURCES AND OTHER DIGITAL MATERIALS

1. Prof. Debasis Samanta, Department of Computer Science & Engineering, I.I.T, Kharagpur, Swayam, NPTEL, https://onlinecourses.nptel.ac.in/noc21_cs03/preview, Last Accessed On: 14/04/ 2024.
2. Evan Jones, Adam Marcus,Eugene Wu Introduction to Programming in Java, MIT OpenCourseWare, Massachusetts Institute of Technology, https://ocw.mit.edu/courses/6-092-introduction-to-programming-in-java-january-iap-2010/, Last Accessed On: 28/05/2024.
3. Prof. Owen Astrachan, Object Oriented Programming in Java, Duke University, https://www.coursera.org/specializations/object-oriented-programming, Last Accessed On: 21/05/2024.
4. Dheeru Mundluru, Java In-Depth: Become a Complete Java Engineer, Udemy, https://www.udemy.com/course/java-in-depth-become-a-complete-java-engineer/, Last Accessed On: 14/05/2024.
5. Prof. Olufisayo Omojokun, Introduction to Object-Oriented Programming with Java I: Foundations and Syntax Basics, Georgia Institute of Technology, edX, https://www.edx.org/certificates/professional-certificate/gtx-introduction-to-object-oriented-programming-with-java, Last Accessed On: 04/05/2024.

23TP3106
LOGIC & REASONING

Course Category	Soft Skill-1	Credits	1
Course Type	Theory	L-T-P	0-0-2
Prerequisites	–	Continuous Eval Semester End Eval Total Marks	100 0 100

COURSE OUTCOMES

1. Think reason logically in any critical situation.
2. Analyze given information to find correct solution.
3. To reduce the mistakes in day to day activities in practical life.
4. Develop time management skills by approaching different shortcut methods.
5. Use mathematical based reasoning to make decisions.
6. Apply logical thinking to solve problems and puzzles in qualifying exams for companies and in other competitive exams.

COURSE CONTENT

UNIT I

1. Series Completion
2. Coding-Decoding
3. Blood Relation
4. Puzzles test

UNIT II

1. Direction sense test
2. Logical Venn diagrams
3. Number test, ranking test
4. Mathematical operations

UNIT III

1. Arithmetical Reasoning
2. Inserting missing character
3. Syllogism
4. Binary logic
5. Data sufficiency

UNIT IV

1. Water images
2. Mirror images
3. Paper folding
4. Paper cutting
5. Embedded Figures
6. Dot situation

7. Cubes & Dice

TEXTBOOKS

1. R. S. Aggarwal, Verbal and non-verbal reasoning, S Chand publication, Revised Edition, 2017, ISBN:81-219-0551-6,
2. Vikramjeeth, Reasoning Guru Verbal & Non-Verbal Reasoning, Multilingual Edition, 2023, ISBN :978-9358706000.

23AI&ML3651
PYTHON PROGRAMMING LAB

Course Category	Skill Enhancement Course	Credits	1
Course Type	Laboratory	L-T-P	0-0-2
Prerequisites	23ES1152 Computer Programming Lab	Continuous Eval Semester End Eval Total Marks	30 70 100

COURSE OUTCOMES

1. Introduce core programming concepts of Python programming language.
2. Demonstrate about Python data structures like Lists, Tuples, Sets and dictionaries
3. Implement Functions, Modules and Regular Expressions in Python Programming and to create practical and contemporary applications using these.

Contribution of Course Outcomes towards achievement of Program Outcomes (1 – Low, 2 - Medium, 3 – High)

CO	PO												PSO		BTL	POI
	1	2	3	4	5	6	7	8	9	10	11	12	1	2		
1	2	2	1											1	1	1.6.1, 1.7.1, 2.5.2, 3.6.1, 3.7.2
2	2	2		2										1	1	1.7.1, 1.5.1, 2.5.1, 2.6.2, 4.4.2, 4.5.1
3	3		2		2									1	1	1.6.1, 1.7.1, 3.6.1, 3.6.3, 3.8.3, 5.4.2

UNIT-I:

History of Python Programming Language, Thrust Areas of Python, Installing Anaconda Python Distribution, Installing and Using Jupiter 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 and except 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 Operators iv) Logical Operators v) Bit wise Operators vi) Ternary Operator vii) Membership Operators viii) 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-II:

Functions: Built-In Functions, Commonly Used Modules, Function Definition and Calling the function, return Statement and void Function, Scope and Lifetime of Variables, Default Parameters, Keyword Arguments, *args and **kwargs, Command Line Arguments. Strings: Creating and Storing Strings, Basic String Operations, Accessing Characters in String by Index Number, String Slicing and Joining, String Methods, Formatting Strings. Lists: Creating Lists, Basic List Operations, Indexing and Slicing in Lists, Built-In Functions Used 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. Addition ii. Insertion iii. Slicing 6. Write a program to perform any 5 built-in functions by taking any list.

UNIT-III:

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 and Slicing in Tuples, Built-In Functions Used on Tuples, Relation between Tuples and Lists, Relation between Tuples and Dictionaries, Using zip() Function, Sets, Set Methods, Frozenset.

Sample Experiments:

1. Write a program to create tuples (name, age, address, college) for at least two members and concatenate the 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-IV:

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, Python os and os.path Modules. Object-Oriented Programming: Classes and Objects, Creating Classes in Python, Creating Objects in Python, Constructor Method, Classes with Multiple Objects, Class Attributes Vs Data Attributes, Encapsulation, Inheritance, Polymorphism.

Sample Experiments:

1. Write a program to sort words in a file and put them in another file. The output file should have only lower-case words, so any upper-case words from source must be lowered.
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 items in 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 to calculate its area and perimeter.
7. Implement subclasses for different shapes like circle, triangle, and square.

UNIT-V:

Introduction to Data Science: Functional Programming, JSON and XML in Python, NumPy with 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 where this list contains at least ten values and convert this dictionary as a pandas data frame and explore the data through the data frame as follows: a) Apply head () function to the pandas data frame b) Perform various data selection operations on Data Frame
7. Select any two columns from the above data frame, and observe the change in one attribute with respect to other attribute with scatter and plot operations in matplotlib

TEXT BOOKS

1. Anuradha A. Puntambekar, Programming and Problem Solving using Python (Fundamentals and Applications), Technical Publication, 1^{st}Edition, 2020.

REFERENCE BOOKS

1. Gowri shankar S, Veena A, Introduction to Python Programming, CRC Press, Special Indian Edition, 2019.
2. S Sridhar, J Indumathi, V M Hariharan, Python Programming, Pearson Education, 2^{nd}Edition, 2024
3. Daniel Liang, Introduction, to Programming Using Python, Y, Pearson Eduction, 1^{st}Edition, 2023.

E-RESOURCES AND OTHER DIGITAL MATERIALS

1. Python for Data Science, AI & Development, https://www.coursera.org/learn/python-for-applied-data-science-ai, Last Accessed On: 19/03/2024.
2. Programming for Everybody (Getting Started with Python), https://www.coursera.org/learn/python?specialization=python#syllabus, Last Accessed On: 05/04/2024.

23AI&ML3352
ADVANCED DATA STRUCTURES & ALGORITHMS LAB

Course Category	Program Core	Credits	1.5
Course Type	Laboratory	L-T-P	0-0-3
Prerequisites	23PC2152A Data Structures Lab	Continuous Eval Semester End Eval Total Marks	30 70 100

COURSE OUTCOMES

1. Implement AVL trees, B-Trees,and Heap Trees, efficient binary trees, M-way search trees, graph traversal and shortest path algorithms.
2. Implement graph-based algorithms, design technique - divide and conquer, and greedy methods for solving complex problems.
3. Implement solutions for problems using dynamic programming and backtracking.
4. Apply Branch and Bound techniques to write programs for different problems.

Contribution of Course Outcomes towards achievement of Program Outcomes (1 – Low, 2 - Medium, 3 – High)

CO	PO												PSO		BTL	POI
	1	2	3	4	5	6	7	8	9	10	11	12	1	2		
1	2	2											3		3	1.2.1, 1.7.1, 2.5.1, 2.5.2, 2.5.3, 2,6.3
2	2	3	3										3		3	1.7.1, 2.5.1, 2.5.2, 2.6.3, 2.6.4, 3.5.1
3	2	3	3										3		3	1.7.1, 2.5.1, 2.5.2, 2.6.3, 2.6.4, 3.5.1
4	2	3	3										3		3	1.7.1, 2.5.1, 2.5.2, 2.6.3, 2.6.4, 3.5.1

Experiments covering the Topics:

1. Operations on AVL trees, B-Trees, Heap Trees
2. Graph Traversals
3. Sorting techniques
4. Minimum cost spanning trees
5. Shortest path algorithms
6. 0/1 Knapsack Problem
7. Travelling Salesperson problem
8. Optimal Binary Search Trees

9. N-Queens Problem
10. Job Sequencing

Sample Programs:

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

TEXTBOOKS

1. Reema Thareja, Data Structures using C, Oxford University Press, 2^{nd} Edition, 2014.
2. Ellis Horowitz, Satraj Sahni and Rajasekharan, Fundamentals of Computer Algorithms, Galgotia Publications Pvt. Ltd, 2^{nd} Edition, 2008.

REFERENCE BOOKS

1. Horowitz Sahni and Anderson-Freed, Fundamentals of Data Structures in C, Universities Press, 2^{nd} Edition, 2008.
2. Richard F.Gilberg & B.A.Forouzan, Data Structures A Pseudo code Approach with C, Cengage Learning, 2^{nd} Edition, 2007.
3. M.T.Goodrich and R.Tomassia, Algorithm Design: Foundations, Analysis and Internet examples, John Wiley and sons, 2^{nd} Edition, 2006.
4. T.H.Cormen, C.E.Leiserson, R.L.Rivest and C.Stein, Introduction to Algorithms, Mit Press, 3^{rd} Edition, 2009.
5. Allen Weiss, Data structures and Algorithm Analysis in C, Pearson Education, 2^{nd} Edition, 2002.

E-RESOURCES AND OTHER DIGITAL MATERIALS

1. Advanced Data Structures Course, https://www.tutorialspoint.com/advanceddatastructures/index.asp, Last Accessed On: 31/5/2024.
2. Computer Algorithms, http://peterindia.net/Algorithms.html, Last Accessed on: 24-06-2024.

23AI&ML3353
OBJECT ORIENTED PROGRAMMING THROUGH JAVA LAB

Course Category	Program Core	Credits	1.5
Course Type	Laboratory	L-T-P	0-0-3
Prerequisites	23ES1152 Computer Programming Lab –	Continuous Eval Semester End Eval Total Marks	30 70 100

COURSE OUTCOMES

1. Apply the basic concepts of object oriented programming.
2. Apply multiple inheritance through interfaces for a given application.
3. Apply exceptions, thread capabilities and handling files on a given application.
4. Apply functional programming and Collections framework for a given application.

Contribution of Course Outcomes towards achievement of Program Outcomes (1 – Low, 2 - Medium, 3 – High)

CO	PO												PSO		BTL	POI
	1	2	3	4	5	6	7	8	9	10	11	12	1	2		
1	2	3													3	1.7.1, 2.5.1, 2.5.2, 2.7.1
2	2	2	3										2	2	3	1.7.1, 2.5.1, 2.5.2, 2.6.3, 2.7.1, 3.5.1, 3.8.2
3	2	2	3										2	2	3	1.7.1, 2.5.1, 2.5.2, 2.6.3, 2.6.4 3.5.1, 3.8.2
4	1	2	3										2	2	3	1.7.1, 2.5.1, 2.5.2, 2.6.3, 2.6.4, 3.5.1, 3.6.2, 3.8.2

COURSE CONTENT

Task 1: Apply fundamentals of Java Data types, Variables, Operators, and Control Statements to a given application.
Task 2: Apply the concepts of Classes and Objects to a given application.
Task 3: Apply the concepts of Arrays to a given application.
Task 4: Apply Inheritance and types of Inheritance of a given application.
Task 5: Use concepts of String and String Tokenizer classes and develop a java application.
Task 6: Use Interfaces and develop a java application.
Task 7: Create a package and access members of a package.

Task 8: Apply the concepts of Method Overloading and Method Overriding.
Task 9: Apply the concepts of Exception Handling.
Task 10: Develop a Java application to copy content from one file to another file using I/O Streams.
Task 11: Apply the concepts of Threads and Multithreading on a given application.
Task12: Apply the concepts of Lambda Expressions, Collections Framework and Stream API.

Lab Projects:
1. Design and develop an automated ballot vote system.
2. Design and develop a banking application.

TEXTBOOKS

1. Herbert Schildt, Java The Complete Reference, Oracle Press, 11^{th} Edition, 2019.

REFERENCE BOOKS

1. Herbert Schildt, Dale Skrien, Java Fundamentals: A Comprehension Introduction, McGraw-Hill Education India Pvt., Special Indian Edition, Ltd, 2017.
2. E Balaguruswamy, Programming with Java, Mc Graw Hill, 4^{th} Edition, 2020.
3. Paul J. Dietel and Dr.Harvey M. Deitel, "Java How to Program", Deitel & Associates Inc.l, 11^{th} Edition, 2018.
4. Timothy Budd, Understanding Object Oriented Programming with Java, Pearson Education, Updated edition, 2013.
5. Kathy Sierra & Bert Bates, Head First Java, Oreilly, 2^{nd} Edition, 2023.

E-RESOURCES AND OTHER DIGITAL MATERIALS

1. Prof. Debasis Samanta, Department of Computer Science & Engineering, I.I.T, Kharagpur, Swayam, NPTEL, https://onlinecourses.nptel.ac.in/, Last Accessed On: 14/04/ 2024.
2. Evan Jones, Adam Marcus, Eugene Wu, Introduction to Programming in Java, MIT OpenCourseWare, Massachusetts Institute of Technology, https://ocw.mit.edu/search/?s=department_course_numbers.sort_coursenum, Last Accessed On: 28/05/2024.
3. Prof. Owen Astrachan, Object Oriented Programming in Java, Duke University, https://www.coursera.org/specializations/object-oriented-programming, Last Accessed On: 21/05/2024.
4. Dheeru Mundluru, Java In-Depth: Become a Complete Java Engineer, Udemy, https://www.udemy.com/course/java-in-depth-become-a-complete-java-engineer/, Last Accessed On: 14/05/2024.
5. Prof. Olufisayo Omojokun, Introduction to OOPS with Java I: Foundations and Syntax Basics, Georgia Institute of Technology, edX, https://www.edx.org/certificates/professional-certificate/gtx-introduction-to-object-oriented-programming-with-java, Last Accessed On: 04/05/2024.