Computer Science and Engineering

Multidisciplinary Minor

Note: These MDMs may be opted by students from any engineering discipline, except for those enrolled in Computer Science and Engineering, Computer Engineering, Computer Science and Engineering (Data Science), Artificial Intelligence and Data Science (AIDS), or allied discipline.

Reference	Sem	Subject Code	Subject Title	Type	Credits
MDM#1	III	3CS205MD / 3KE205MD / 3DS205MD / 3AD205MD	Foundations of Computing & Programming	Theory	2
MDM#2	IV	4CS214MD	Data Structures and Problem Solving	Theory	2
MDM#3	V	5CS227MD	Database Management Systems	Theory	2
MDM#4	V	5CS228MD	Web Technologies & Development	Theory	2
MDM#5	V	5CS229ML	Web Technologies Lab	Practical	1
MDM#6	VI	6CS239MD	Cloud Computing & Virtualization	Theory	2
MDM#7	VII	7CS307MD	Computational Thinking & Algorithmic Problem Solving	Theory	2
MDM#8	VII	7CS308ML	Interdisciplinary Computing Project Lab	Practical	1

Subject wise Syllabus

SEM III

Subject Code: 3CS205MD / 3KE205MD / 3DS205MD / 3AD205MD
Subject Title: Foundations of Computing & Programming
Type: Theory | **Credits:** 2
Total Contact Hours: 22–25 hours

Prerequisite: Basic familiarity with mathematics and logical reasoning.

Course Objectives:

- To introduce fundamental concepts of computing and its relevance across engineering domains.
- To develop problem-solving skills using algorithmic approaches.
- To provide exposure to basic programming constructs using a high-level language (Python).
- To build foundational understanding required for advanced computational courses.

Course Outcomes (COs):

1. **CO1:** Understand computing systems and problem-solving logic. (Unit 1)
2. **CO2:** Apply algorithmic thinking to solve simple problems. (Unit 2)
3. **CO3:** Implement basic programs using control structures and I/O operations. (Unit 3)

Unit-wise Syllabus:

Unit 1: Basics of Computing & Programming (7 hrs)
Introduction to Computers and Programming Languages, Components of a Computer System and Software Types, Number Systems (Binary, Decimal, Hexadecimal) and Conversions, Basics of Algorithms and Flowcharts, Introduction to Python: Syntax, Variables, Data Types.

Unit 2: Control Structures and Functions (8 hrs)
Conditional Statements: if, if-else, nested if, Looping Constructs: for, while, Introduction to Functions and Modular Programming, Basic Math Operations and Built-in Functions in Python.

Unit 3: Input-Output and Basic Problem Solving (7 hrs)
Input from User and Output Formatting, Lists and String Handling, Simple Problem-Solving , Examples using Loops and Conditions, Writing and Executing Python Scripts.

CO-PO Mapping Table:

CO \ PO	PO1	PO2	PO4	PO5	PO10	PO12
CO1	3	2				1
CO2	2	3	2			2
CO3		3	2	3	2	3

Textbook(s):

Reema Thareja, "Python Programming: Using Problem Solving Approach," Oxford University Press, 2017.

Reference Books:

1. Allen B. Downey, "Think Python: How to Think Like a Computer Scientist," O'Reilly Media, 2nd Edition, 2015.
2. Zelle, John M., "Python Programming: An Introduction to Computer Science," Franklin, Beedle & Associates, 2017.
3. Anita Goel, "Computer Fundamentals," Pearson Education, 2010.
4. Charles Severance, "Python for Everybody: Exploring Data Using Python 3," CreateSpace, 2016.
5. Yashavant Kanetkar, "Let Us Python," BPB Publications, 2020.

==XX==

SEM IV

Subject Code: 4CS214MD
Subject Title: Data Structures and Problem Solving
Type: Theory | **Credits**: 2
Total Contact Hours: 22–23 hours

Prerequisite: Basic understanding of programming (preferably in C or Python).

Course Objectives:

- To introduce fundamental data structures and their applications.
- To develop problem-solving skills through efficient use of data structures.
- To improve algorithmic thinking for common computational tasks.
- To expose students to practical scenarios where data structures are applied in engineering contexts.

Course Outcomes (COs):

1. **CO1**: Understand and differentiate various data structures and their use cases. (Unit 1)
2. **CO2**: Apply linear and non-linear data structures in solving engineering problems. (Unit 2)
3. **CO3**: Analyze algorithm performance and implement solutions using appropriate structures. (Unit 3)

Unit-wise Syllabus:

Unit 1: Introduction to Data Structures (7 hrs)
Concept and Classification of Data Structures, Abstract Data Types (ADT), Arrays, Strings, and Linked Lists, Applications: Polynomial Representation, Sparse Matrices.

Unit 2: Linear and Non-linear Structures (8 hrs)
Stacks and Queues: Operations and Applications (Infix to Postfix, Recursion, Job Scheduling) Trees: Binary Trees, Tree Traversals, Binary Search Trees, Applications in Expression Trees, Directory Structures.

Unit 3: Searching, Sorting & Hashing (7 hrs)
Searching Techniques: Linear and Binary Search, Sorting Algorithms: Bubble, Selection, Insertion, Merge Sort, Hashing and Collision Resolution Techniques, Time and Space Complexity Basics.

CO-PO Mapping Table:

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO12
CO1	3	2				
CO2	2	3	3	2	2	2
CO3	2	3	3	3	3	2

Textbook(s):

1. Seymour Lipschutz, "Data Structures", Schaum's Outline Series, McGraw Hill Education.
2. Reema Thareja, "Data Structures Using C", Oxford University Press, 2nd Edition, 2014.

Reference Books:

1. Ellis Horowitz, Sartaj Sahni, Susan Anderson-Freed, "Fundamentals of Data Structures in C", Universities Press, 2008.
2. Jean-Paul Tremblay & Paul G. Sorenson, "An Introduction to Data Structures with Applications", Tata McGraw-Hill.
3. Narasimha Karumanchi, "Data Structures and Algorithms Made Easy", CareerMonk Publications.
4. Robert Lafore, "Data Structures and Algorithms in C++", Pearson Education.
5. Mark Allen Weiss, "Data Structures and Algorithm Analysis in C", Pearson Education.
6. Thomas H. Cormen et al., "Introduction to Algorithms", MIT Press (for higher interest).

==XX==

Computer Engineering

Multidisciplinary Minor

Note: These MDMs may be opted by students from any engineering discipline, except for those enrolled in Computer Science and Engineering, Computer Engineering, Computer Science and Engineering (Data Science), Artificial Intelligence and Data Science (AIDS), or allied discipline.

Reference	Sem	Subject Code	Subject Title	Type	Credits
MDM#1	III	3KE205MD / 3CS205MD / 3DS205MD / 3AD205MD	Foundations of Computing & Programming	Theory	2
MDM#2	IV	4KE214MD	Operating Systems Fundamentals	Theory	2
MDM#3	V	5KE227MD	Software Engineering Principles	Theory	2
MDM#4	V	5KE228MD	Mobile Application Development	Theory	2
MDM#5	V	5KE229ML	Mobile App Development Lab (Android/Kotlin)	Practical	1
MDM#6	VI	6KE239MD	DevOps and Software Deployment Practices	Theory	2
MDM#7	VII	7KE307MD	Full Stack Web Development	Theory	2
MDM#8	VII	7KE308ML	Software Project Lab	Practical	1

Subject wise Syllabus

SEM III

Subject Code: 3KE205MD / 3CS205MD / 3DS205MD / 3AD205MD
Subject Title: Foundations of Computing & Programming
Type: Theory | **Credits:** 2
Total Contact Hours: 22–25 hours

Prerequisite: Basic familiarity with mathematics and logical reasoning.

Course Objectives:

- To introduce fundamental concepts of computing and its relevance across engineering domains.
- To develop problem-solving skills using algorithmic approaches.
- To provide exposure to basic programming constructs using a high-level language (Python).
- To build foundational understanding required for advanced computational courses.

Course Outcomes (COs):

1. **CO1:** Understand computing systems and problem-solving logic. (Unit 1)
2. **CO2:** Apply algorithmic thinking to solve simple problems. (Unit 2)
3. **CO3:** Implement basic programs using control structures and I/O operations. (Unit 3)

Unit-wise Syllabus:

Unit 1: Basics of Computing & Programming (7 hrs)
Introduction to Computers and Programming Languages, Components of a Computer System and Software Types, Number Systems (Binary, Decimal, Hexadecimal) and Conversions, Basics of Algorithms and Flowcharts, Introduction to Python: Syntax, Variables, Data Types.

Unit 2: Control Structures and Functions (8 hrs)
Conditional Statements: if, if-else, nested if, Looping Constructs: for, while, Introduction to Functions and Modular Programming, Basic Math Operations and Built-in Functions in Python.

Unit 3: Input-Output and Basic Problem Solving (7 hrs)
Input from User and Output Formatting, Lists and String Handling, Simple Problem-Solving , Examples using Loops and Conditions, Writing and Executing Python Scripts.

CO-PO Mapping Table:

CO \ PO	PO1	PO2	PO4	PO5	PO10	PO12
CO1	3	2				1
CO2	2	3	2			2
CO3		3	2	3	2	3

Textbook(s):

Reema Thareja, "Python Programming: Using Problem Solving Approach," Oxford University Press, 2017.

Reference Books:

1. Allen B. Downey, "Think Python: How to Think Like a Computer Scientist," O'Reilly Media, 2nd Edition, 2015.
2. Zelle, John M., "Python Programming: An Introduction to Computer Science," Franklin, Beedle & Associates, 2017.
3. Anita Goel, "Computer Fundamentals," Pearson Education, 2010.
4. Charles Severance, "Python for Everybody: Exploring Data Using Python 3," CreateSpace, 2016.
5. Yashavant Kanetkar, "Let Us Python," BPB Publications, 2020.

==XX==

SEM IV

Subject Code: 4CE205MD
Subject Title: Operating Systems Fundamentals
Type: Theory | **Credits**: 2
Total Contact Hours: 22–25 hours

Prerequisite: Familiarity with computer hardware basics and programming fundamentals.

Course Objectives:

- To introduce students to the fundamentals of operating systems and their role in computing.
- To explain the key functions of an OS including process, memory, and file management.
- To describe the basic concepts of multitasking, scheduling, and resource allocation.
- To provide a foundation for understanding system-level operations and future system software courses.

Course Outcomes (COs):

1. **CO1:** Understand the basic components and functions of an operating system. (Unit 1)
2. **CO2:** Explain concepts of process management and CPU scheduling. (Unit 2)
3. **CO3:** Demonstrate an understanding of memory and file system management. (Unit 3)

Unit-wise Syllabus:

Unit 1: Overview of Operating Systems (7 hrs)
Definition and Functions of Operating System, Types of Operating Systems: Batch, Multitasking, Time-Sharing, Real-Time, Mobile OS; Components of OS – Kernel, Shell, System Calls; System Boot Process; Examples of OS (Windows, Linux, Android).

Unit 2: Process Management and Scheduling (8 hrs)
Concept of Process, Process States and PCB; Threads and Multithreading; CPU Scheduling: FCFS, SJF, Round Robin (RR); Introduction to Deadlocks – Necessary Conditions, Deadlock Prevention and Recovery (Basic Concepts).

Unit 3: Memory & File Management (7 hrs)
Memory Hierarchy and Management Techniques: Paging, Segmentation (Conceptual Overview); Introduction to Virtual Memory; File Systems and Operations: Creation, Reading, Writing, Deletion; Directory Structures and File Allocation (Concept Only); Introduction to Disk Scheduling (FCFS, SSTF).

CO-PO Mapping Table:

CO \ PO	PO1	PO2	PO4	PO5	PO10	PO12
CO1	3	2				1
CO2	2	3	2			2
CO3		3	3	2	2	3

Textbook(s):

Silberschatz, Galvin, and Gagne, *Operating System Concepts*, 9th Edition, Wiley, 2013.

Reference Books:

1. Andrew S. Tanenbaum, Modern Operating Systems, Pearson, 4th Edition, 2014.
2. William Stallings, Operating Systems: Internals and Design Principles, Pearson, 8th Edition, 2018.
3. Achyut Godbole and Atul Kahate, Operating Systems, McGraw Hill Education, 4th Edition, 2017.
4. D.M. Dhamdhere, Operating Systems: A Concept-Based Approach, McGraw Hill, 2012.
5. Milan Milenkovic, Operating Systems: Concepts and Design, McGraw Hill, 2001.
6. G. Nutt, Operating Systems: A Modern Perspective, Pearson, 2nd Edition, 2003.

==XX==

Computer Science and Engineering (Data Science)

Multidisciplinary Minor

Note: These MDMs may be opted by students from any engineering discipline, except for those enrolled in Computer Science and Engineering, Computer Engineering, Computer Science and Engineering (Data Science), Artificial Intelligence and Data Science (AIDS), or allied discipline.

Reference	Sem	Subject Code	Subject Title	Type	Credits
MDM#1	III	3DS205MD / 3CS205MD / 3KE205MD / 3AD205MD	Foundations of Computing & Programming	Theory	2
MDM#2	IV	4DS214MD	Foundations of Data Science & Statistics	Theory	2
MDM#3	V	5DS227MD	Data Wrangling & Preprocessing Techniques	Theory	2
MDM#4	V	5DS228MD	Data Visualization Techniques	Theory	2
MDM#5	V	5DS229ML	Data Analytics Lab	Practical	1
MDM#6	VI	6DS239MD	Applied Machine Learning for Engineers	Theory	2
MDM#7	VII	7DS307MD	Predictive Analytics & Decision Support	Theory	2
MDM#8	VII	7DS308ML	Data Science Project Lab	Practical	1

Subject wise Syllabus

SEM III

Subject Code: 3DS205MD / 3CS205MD / 3KE205MD / 3AD205MD
Subject Title: Foundations of Computing & Programming
Type: Theory | **Credits:** 2
Total Contact Hours: 22–25 hours

Prerequisite: Basic familiarity with mathematics and logical reasoning.

Course Objectives:

- To introduce fundamental concepts of computing and its relevance across engineering domains.
- To develop problem-solving skills using algorithmic approaches.
- To provide exposure to basic programming constructs using a high-level language (Python).
- To build foundational understanding required for advanced computational courses.

Course Outcomes (COs):

1. **CO1:** Understand computing systems and problem-solving logic. (Unit 1)
2. **CO2:** Apply algorithmic thinking to solve simple problems. (Unit 2)
3. **CO3:** Implement basic programs using control structures and I/O operations. (Unit 3)

Unit-wise Syllabus:

Unit 1: Basics of Computing & Programming (7 hrs)
Introduction to Computers and Programming Languages, Components of a Computer System and Software Types, Number Systems (Binary, Decimal, Hexadecimal) and Conversions, Basics of Algorithms and Flowcharts, Introduction to Python: Syntax, Variables, Data Types.

Unit 2: Control Structures and Functions (8 hrs)
Conditional Statements: if, if-else, nested if, Looping Constructs: for, while, Introduction to Functions and Modular Programming, Basic Math Operations and Built-in Functions in Python.

Unit 3: Input-Output and Basic Problem Solving (7 hrs)
Input from User and Output Formatting, Lists and String Handling, Simple Problem-Solving , Examples using Loops and Conditions, Writing and Executing Python Scripts.

CO-PO Mapping Table:

CO \ PO	PO1	PO2	PO4	PO5	PO10	PO12
CO1	3	2				1
CO2	2	3	2			2
CO3		3	2	3	2	3

Textbook(s):

Reema Thareja, "Python Programming: Using Problem Solving Approach," Oxford University Press, 2017.

Reference Books:

1. Allen B. Downey, "Think Python: How to Think Like a Computer Scientist," O'Reilly Media, 2nd Edition, 2015.
2. Zelle, John M., "Python Programming: An Introduction to Computer Science," Franklin, Beedle & Associates, 2017.
3. Anita Goel, "Computer Fundamentals," Pearson Education, 2010.
4. Charles Severance, "Python for Everybody: Exploring Data Using Python 3," CreateSpace, 2016.
5. Yashavant Kanetkar, "Let Us Python," BPB Publications, 2020.

==XX==

SEM IV

Subject Code: 4DS214MD
Subject Title: Foundations of Data Science & Statistics
Type: Theory | **Credits**: 2
Total Contact Hours: 22–25 hours

Prerequisite: Basic knowledge of mathematics, statistics, and computer operations.

Course Objectives:

- To introduce the fundamental concepts and workflow of Data Science.
- To build understanding of essential descriptive statistics and probability theory.
- To equip students with data interpretation and visualization skills using simple tools.
- To develop a statistical foundation required for data-driven engineering applications.

Course Outcomes (COs):

1. **CO1:** Understand the concepts, tools, and processes of the data science lifecycle. (Unit 1)
2. **CO2:** Apply descriptive statistics and probability for data summarization and inference. (Unit 2)
3. **CO3:** Perform data analysis and visualization using basic tools. (Unit 3)

Unit-wise Syllabus

Unit 1: Introduction to Data Science (7 hrs)
What is Data Science? Components and Lifecycle; Role of a Data Scientist; Types of Data (Structured, Unstructured, Semi-structured); Real-world Applications of Data Science; Overview of Data Collection and Preparation; Introduction to Python or Excel as data analysis tools.

Unit 2: Statistics and Probability for Data Science (8 hrs)
Descriptive Statistics: Measures of Central Tendency (Mean, Median, Mode), Dispersion (Range, Variance, Standard Deviation); Introduction to Probability: Basic Rules, Conditional Probability, Bayes' Theorem; Overview of Probability Distributions: Normal, Binomial, Poisson (Conceptual only).

Unit 3: Data Analysis & Visualization (7 hrs)
Exploratory Data Analysis (EDA): Identifying Trends and Outliers; Introduction to Data Cleaning (Missing Values, Duplicates); Data Visualization using Charts (Bar, Histogram, Pie, Boxplots); Introduction to Tools like Matplotlib/Excel/Power BI; Case Studies from Healthcare, Retail, or Education.

CO-PO Mapping Table

CO \ PO	PO1	PO2	PO4	PO5	PO10	PO12
CO1	3	2				2
CO2	2	3	2			2
CO3		3	3	3	2	3

Textbook(s):

Joel Grus, *Data Science from Scratch*, O'Reilly Media, 2nd Edition, 2019.

Reference Books:

1. Cathy O'Neil and Rachel Schutt, Doing Data Science, O'Reilly Media, 2014.
2. Wes McKinney, Python for Data Analysis, O'Reilly Media, 2nd Edition, 2017.
3. David Spiegelhalter, The Art of Statistics: Learning from Data, Penguin, 2019.
4. Allen B. Downey, Think Stats: Probability and Statistics for Programmers, O'Reilly Media, 2014.
5. Charles Wheelan, Naked Statistics: Stripping the Dread from the Data, W. W. Norton, 2013.
6. G. Jay Kerns, Introduction to Probability and Statistics Using R, Free Online Book, 2010.

==XX==

Artificial Intelligence and Data Science

Multidisciplinary Minor

Note: These MDMs may be opted by students from any engineering discipline, except for those enrolled in Computer Science and Engineering, Computer Engineering, Computer Science and Engineering (Data Science), Artificial Intelligence and Data Science (AIDS), or allied discipline.

Reference	Sem	Subject Code	Subject Title	Type	Credits
MDM#1	III	3AD205MD / 3CS205MD / 3KE205MD / 3DS205MD	Foundations of Computing & Programming	Theory	2
MDM#2	IV	4AD214MD	Essentials of Artificial Intelligence	Theory	2
MDM#3	V	5AD227MD	Introduction to Machine Learning	Theory	2
MDM#4	V	5AD228MD	AI for Engineering Applications	Theory	2
MDM#5	V	5AD229ML	Machine Learning Lab	Practical	1
MDM#6	VI	6AD239MD	Deep Learning Foundations	Theory	2
MDM#7	VII	7AD307MD	AI in Robotics, Healthcare, and Smart Systems	Theory	2
MDM#8	VII	7AD308ML	AI Tools and Frameworks Lab	Practical	1

Subject wise Syllabus

SEM III

Subject Code: 3AD205MD / 3CS205MD / 3KE205MD / 3DS205MD
Subject Title: Foundations of Computing & Programming
Type: Theory | **Credits:** 2
Total Contact Hours: 22–25 hours

Prerequisite: Basic familiarity with mathematics and logical reasoning.

Course Objectives:

- To introduce fundamental concepts of computing and its relevance across engineering domains.
- To develop problem-solving skills using algorithmic approaches.
- To provide exposure to basic programming constructs using a high-level language (Python).
- To build foundational understanding required for advanced computational courses.

Course Outcomes (COs):

1. **CO1:** Understand computing systems and problem-solving logic. (Unit 1)
2. **CO2:** Apply algorithmic thinking to solve simple problems. (Unit 2)
3. **CO3:** Implement basic programs using control structures and I/O operations. (Unit 3)

Unit-wise Syllabus:

Unit 1: Basics of Computing & Programming (7 hrs)
Introduction to Computers and Programming Languages, Components of a Computer System and Software Types, Number Systems (Binary, Decimal, Hexadecimal) and Conversions, Basics of Algorithms and Flowcharts, Introduction to Python: Syntax, Variables, Data Types.

Unit 2: Control Structures and Functions (8 hrs)
Conditional Statements: if, if-else, nested if, Looping Constructs: for, while, Introduction to Functions and Modular Programming, Basic Math Operations and Built-in Functions in Python.

Unit 3: Input-Output and Basic Problem Solving (7 hrs)
Input from User and Output Formatting, Lists and String Handling, Simple Problem-Solving , Examples using Loops and Conditions, Writing and Executing Python Scripts.

CO-PO Mapping Table:

CO \ PO	PO1	PO2	PO4	PO5	PO10	PO12
CO1	3	2				1
CO2	2	3	2			2
CO3		3	2	3	2	3

Textbook(s):

Reema Thareja, "Python Programming: Using Problem Solving Approach," Oxford University Press, 2017.

Reference Books:

1. Allen B. Downey, "Think Python: How to Think Like a Computer Scientist," O'Reilly Media, 2nd Edition, 2015.
2. Zelle, John M., "Python Programming: An Introduction to Computer Science," Franklin, Beedle & Associates, 2017.
3. Anita Goel, "Computer Fundamentals," Pearson Education, 2010.
4. Charles Severance, "Python for Everybody: Exploring Data Using Python 3," CreateSpace, 2016.
5. Yashavant Kanetkar, "Let Us Python," BPB Publications, 2020.

==XX==

SEM IV

Subject Code: 4AD214MD
Subject Title: Essentials of Artificial Intelligence
Type: Theory | **Credits**: 2
Total Contact Hours: 22–25 hours

Prerequisite: Basic understanding of mathematics, logic, and programming fundamentals.

Course Objectives:

- To introduce fundamental concepts and domains of Artificial Intelligence (AI).
- To explore classical AI techniques for problem-solving and search.
- To familiarize students with basic applications of AI in various fields.
- To build foundational awareness required for more advanced AI and ML topics.

Course Outcomes (COs):

1. **CO1:** Understand the history, goals, and scope of Artificial Intelligence. (Unit 1)
2. **CO2:** Apply search strategies and reasoning techniques to solve basic problems. (Unit 2)
3. **CO3:** Explore real-world applications and ethical considerations of AI. (Unit 3)

Unit-wise Syllabus

Unit 1: Introduction to Artificial Intelligence (7 hrs)
Definition and Evolution of AI, Goals and Components of AI, Branches and Subfields (ML, NLP, Vision, Robotics), Applications in Engineering, Healthcare, and Finance; Intelligent Agents: Types and Environments, Task Environments, PEAS Framework.

Unit 2: Problem Solving and Knowledge Representation (8 hrs)
Concept of State Space and Search Strategies, Uninformed Search: BFS, DFS, Iterative Deepening; Introduction to Heuristics and A* Search; Basics of Knowledge Representation: Propositional Logic, Rule-Based Systems; Use Cases in Diagnostics and Planning.

Unit 3: Applications and Ethics in AI (7 hrs)
AI in the Real World: Use Cases in Chatbots, Autonomous Vehicles, Personal Assistants, Smart Cities; Overview of Machine Learning vs AI; Bias and Fairness in AI; Ethical and Societal Considerations; Career Opportunities in AI for Non-CS Backgrounds.

CO-PO Mapping Table

CO \ PO	PO1	PO2	PO4	PO5	PO6	PO8	PO12
CO1	3				2		2
CO2	2	3	2				2
CO3		2		2	2	3	3

Textbook(s):

Stuart Russell & Peter Norvig, *Artificial Intelligence: A Modern Approach*, Pearson Education, 4th Edition, 2020.

Reference Books:

1. Elaine Rich, Kevin Knight, Shivashankar B. Nair, Artificial Intelligence, McGraw-Hill Education, 3rd Edition, 2008.
2. Dan W. Patterson, Introduction to Artificial Intelligence and Expert Systems, Pearson, 2005.
3. Ethem Alpaydin, Introduction to Machine Learning, MIT Press, 3rd Edition, 2014.
4. John Haugeland, Artificial Intelligence: The Very Idea, MIT Press, 1985.
5. Nils J. Nilsson, The Quest for Artificial Intelligence, Cambridge University Press, 2009.
6. Toby Walsh, Machines Behaving Badly: The Morality of AI, Basic Books, 2022.

==XX==