Haldia Institute of Technology (Autonomous)

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

PROGRAM OUTCOMES (ALIGNED WITH GRADUATE ATTRIBUTES) (POs)

At the end of this program, graduates 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: 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.

Curriculum Structure

<table>
<thead>
<tr>
<th>Sl. No.</th>
<th>Type of Course</th>
<th>Code</th>
<th>Course Title</th>
<th>Hours per week</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Theory</td>
<td>ESC-AIML-301</td>
<td>Digital Electronics</td>
<td>3 0 0</td>
<td>3</td>
</tr>
<tr>
<td>1</td>
<td>Engineering Science Course</td>
<td>PCC-AIML-301</td>
<td>Data Structure and</td>
<td>3 1 0</td>
<td>4</td>
</tr>
<tr>
<td>Sl. No.</td>
<td>Type of Course</td>
<td>Code</td>
<td>Course Title</td>
<td>Hours per week</td>
<td>Credits</td>
</tr>
<tr>
<td>--------</td>
<td>---------------------------------------------------</td>
<td>---------------</td>
<td>--------------------------------------------------</td>
<td>----------------</td>
<td>---------</td>
</tr>
<tr>
<td>3</td>
<td>Professional Core Course</td>
<td>PCC-AIML-302</td>
<td>Algorithms</td>
<td>3 1 0 4</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Basic Science Course</td>
<td>BSC-AIML-301</td>
<td>Linear Algebra</td>
<td>3 0 0 3</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Humanities Social Science including Management</td>
<td>HSMC-AIML-301</td>
<td>Economics for Engineers</td>
<td>2 0 0 2</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Engineering Science Course</td>
<td>ESC-AIML-391</td>
<td>Digital Electronics Lab</td>
<td>0 0 3 1.5</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Professional Core Course</td>
<td>PCC-AIML-391</td>
<td>Data Structure and Algorithms Lab</td>
<td>0 0 3 1.5</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Professional Core Course</td>
<td>PCC-AIML-392</td>
<td>Computer Architecture Lab</td>
<td>0 0 3 1.5</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Professional Core Course</td>
<td>PCC-AIML-393</td>
<td>IT Workshop (Python)</td>
<td>0 0 3 1.5</td>
<td></td>
</tr>
</tbody>
</table>

**Total Credits**: 22

<table>
<thead>
<tr>
<th>Sl. No.</th>
<th>Type of Course</th>
<th>Code</th>
<th>Course Title</th>
<th>Hours per week</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Engineering Science Course</td>
<td>ESC-AIML-401</td>
<td>Probability and Statistics</td>
<td>3 0 0 3</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>Professional Core Course</td>
<td>PCC-AIML-401</td>
<td>Object Oriented Programming &amp; Java</td>
<td>3 0 0 3</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>Professional Core Course</td>
<td>PCC-AIML-402</td>
<td>Formal Language and Automata Theory</td>
<td>3 0 0 3</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>Professional Core Course</td>
<td>PCC-AIML-403</td>
<td>Design and Analysis of Algorithm</td>
<td>3 1 0 4</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>Engineering Science Course</td>
<td>ESC-AIML-402</td>
<td>Signals and Systems</td>
<td>3 0 0 3</td>
<td>3</td>
</tr>
<tr>
<td>6</td>
<td>Mandatory Course</td>
<td>MC-AIML-401</td>
<td>Environmental Sciences</td>
<td>2 0 0 0</td>
<td>0</td>
</tr>
<tr>
<td>7</td>
<td>Professional Core Course</td>
<td>PCC-AIML-491</td>
<td>Object Oriented Programming &amp; Java Lab</td>
<td>0 0 3 1.5</td>
<td>1.5</td>
</tr>
<tr>
<td>8</td>
<td>Professional Core Course</td>
<td>PCC-AIML-492</td>
<td>Design and Analysis of Algorithm Lab</td>
<td>0 0 3 1.5</td>
<td>1.5</td>
</tr>
</tbody>
</table>

**Total Credits**: 19

<table>
<thead>
<tr>
<th>Sl. No.</th>
<th>Type of Course</th>
<th>Code</th>
<th>Course Title</th>
<th>Hours per week</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Basic Science Course</td>
<td>BSC-AIML-501</td>
<td>Discrete Mathematics</td>
<td>3 0 0 3</td>
<td>3</td>
</tr>
<tr>
<td>Sl. No.</td>
<td>Type of Course</td>
<td>Code</td>
<td>Course Title</td>
<td>Hours per week</td>
<td>Credits</td>
</tr>
<tr>
<td>--------</td>
<td>----------------</td>
<td>-------------------</td>
<td>-------------------------------------------</td>
<td>----------------</td>
<td>---------</td>
</tr>
<tr>
<td></td>
<td>Professional Core Course</td>
<td>PCC-AIML-501</td>
<td>Operating System</td>
<td>3 0 0 3</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Professional Core Course</td>
<td>PCC-AIML-502</td>
<td>Database Management Systems</td>
<td>3 0 0 3</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Professional Core Course</td>
<td>PCC-AIML-503</td>
<td>Machine Learning Foundations</td>
<td>3 0 0 3</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Professional Core Course</td>
<td>PCC-AIML-504</td>
<td>Computer Networks</td>
<td>3 0 0 3</td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>Professional Core Course</td>
<td>PCC-AIML-505</td>
<td>Artificial Intelligence</td>
<td>3 0 0 3</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Practical</td>
<td>PCC-AIML-591</td>
<td>Operating System Lab</td>
<td>0 0 3 1.5</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Professional Core Course</td>
<td>PCC-AIML-592</td>
<td>Database Management Systems Lab</td>
<td>0 0 3 1.5</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Professional Core Course</td>
<td>PCC-AIML-593</td>
<td>Machine Learning Foundations Lab</td>
<td>0 0 3 1.5</td>
<td></td>
</tr>
</tbody>
</table>

**Total Credits**

22.5

**Semester VI (Third year)**

<table>
<thead>
<tr>
<th>Sl. No.</th>
<th>Type of Course</th>
<th>Code</th>
<th>Course Title</th>
<th>Hours per week</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Professional Core Course</td>
<td>PCC-AIML-601</td>
<td>Machine Learning Applications</td>
<td>2 0 0 2</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Professional Core Course</td>
<td>PCC-AIML-602</td>
<td>Deep Learning</td>
<td>3 0 0 3</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Humanities and Social Sciences including Management</td>
<td>HSMC-AIML-601</td>
<td>Human Values and Professional Ethics</td>
<td>3 0 0 3</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Professional Elective - I</td>
<td>PEC-AIML-601 (A/B/C/D)</td>
<td>Soft Computing Cloud Computing Pattern Recognition Graph Theory</td>
<td>3 0 0 3</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Professional Elective - II</td>
<td>PEC-AIML-602 (A/B/C/D)</td>
<td>Big Data Analytics Data Mining Distributed System Digital Signal Processing</td>
<td>3 0 0 3</td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>Professional Core Course</td>
<td>PCC-AIML-691</td>
<td>Machine Learning Applications Lab</td>
<td>0 0 3 1.5</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>Professional Core Course</td>
<td>PCC-AIML-692</td>
<td>Artificial Intelligence Lab</td>
<td>0 0 3 1.5</td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>Professional Core Course</td>
<td>PCC-AIML-693</td>
<td>Computer Networks Lab</td>
<td>0 0 3 1.5</td>
<td></td>
</tr>
<tr>
<td>Sl. No.</td>
<td>Type of Course</td>
<td>Code</td>
<td>Course Title</td>
<td>Hours per week</td>
<td>Credits</td>
</tr>
<tr>
<td>--------</td>
<td>---------------</td>
<td>-----------------</td>
<td>-------------------------------------------------------------------------------</td>
<td>----------------</td>
<td>---------</td>
</tr>
<tr>
<td>1</td>
<td>Professional Elective Course - III</td>
<td>PEC-AIML-701 (A/B/C/D)</td>
<td>Social Network Analysis, Computer Vision, Software Engineering, Data Warehousing</td>
<td>3 0 0</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>Professional Elective Course - IV</td>
<td>PEC-AIML-702 (A/B/C/D)</td>
<td>Ecommerce and ERP, Information Theory, Coding, Data Visualization, Mobile Computing</td>
<td>3 0 0</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>Open Elective Course - II</td>
<td>OEC-AIML-701 (A/B/C/D)</td>
<td>Internet of Things, Bio Informatics, Robotics, Compiler Design</td>
<td>3 0 0</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>Humanities and Social Sciences Including Management</td>
<td>HSMC-AIML-701</td>
<td>Principles of Management</td>
<td>3 0 0</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td>Project</td>
<td>PROJ-AIML-781</td>
<td>Project I</td>
<td>0 0 10</td>
<td>5</td>
</tr>
<tr>
<td>6</td>
<td>Summer Internship</td>
<td>SI-AIML-781</td>
<td>Internship - I</td>
<td></td>
<td>3</td>
</tr>
</tbody>
</table>

**Total Credits**: 20

<table>
<thead>
<tr>
<th>Sl. No.</th>
<th>Type of Course</th>
<th>Code</th>
<th>Course Title</th>
<th>Hours per week</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Professional Elective Course - V</td>
<td>PEC-AIML-801 (A/B/C/D)</td>
<td>Natural Language Processing, Ethical Issues in AI, Digital Image Processing, Applications of AI in Biomedical Engineering</td>
<td>3 0 0</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>Open Elective Course - III</td>
<td>OEC-AIML-801 (A/B/C/D)</td>
<td>Operation Research, Economic Policies in India, Microelectronics and VLSI, Quantum Computing</td>
<td>3 0 0</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td>Open Elective Course - IV</td>
<td>OEC-AIML-802 (A/B/C/D)</td>
<td>Organizational Behaviour, Human Resource Management, Research Methodology</td>
<td>3 0 0</td>
<td>3</td>
</tr>
</tbody>
</table>
ESC-AIML-301: Digital Electronics
Contacts: 3L per week Credits: 3

Course Outcomes (COs):

At the end of this course, students will demonstrate the ability to
1. Understand working of logic families and logic gates.
2. Convert a number from one base system to another.
3. Understand Boolean algebra and represent digital circuits from Boolean expressions.
4. Design and implement many Combinational circuits.
5. Design and implement various Sequential logic circuits.
6. Understand the process of Analog to Digital conversion and Digital to Analog conversion.

Prerequisites:
1. Number Systems

Detailed Content:

<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Digital signals, digital circuits, AND, OR, NOT, NAND, NOR and Exclusive-OR operations, Binary Number System &amp; Boolean Algebra (recapitulation); BCD, ASCII, EBDIC, Gray codes and their conversions; Signed binary number representation with 1’s and 2’s complement methods, Binary arithmetic, Venn diagram,</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>Boolean algebra (recapitulation); Representation in SOP and POS forms; Minimization of logic expressions by algebraic method. K-map representation, simplification of logic functions using K-map, minimization of logical functions. Don’t care conditions.</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>Combinational Circuits - Adder and Subtractor circuits (half &amp; full adder &amp; subtractor); Encoder, Decoder, Comparator, Multiplexer, De-multiplexer and Parity Generator. Error detecting and correcting codes. Adders, Subtractors, BCD arithmetic, carry look ahead adder, serial adder</td>
<td>7</td>
</tr>
<tr>
<td>4</td>
<td>Sequential Circuits - Basic Flip-flop &amp; Latch, Flip-flops -SR, JK, D, T and JK Master-slave Flip Flops, Registers (SISO, SIPO, PIPO, PISO) Ring counter, Johnson counter Basic concept of Synchronous and Asynchronous counters (detail design of circuits excluded), Design of Mod N Counter. Counters design using flip-flops</td>
<td>8</td>
</tr>
<tr>
<td>5</td>
<td>A/D and D/A conversion techniques – Basic concepts (D/A :R-2-R only [2L] A/D: successive approximation [2L]) Logic families- TTL, ECL, MOS and CMOS - basic concepts. (2L)</td>
<td>6</td>
</tr>
</tbody>
</table>
Text book and Reference books:

3. Analog Electronics, L.K. Maheswari, Laxmi Publications (AICTE Recommended -2018)
4. Principles of Electronic Devices & circuits—B L Thereja & Sedha—S Chand
5. Digital Electronics – Kharate – Oxford
10. P.Raja- Digital Electronics- Scitech Publications
11. Morries Mano- Digital Logic Design- PHI
15. Tocci, Widmer, Moss- Digital Systems,9/e- Pearson
17. Leach & Malvino—Digital Principles & Application, 5/e, McGraw Hill

PCC-AIML-301: Data Structure and Algorithms

Lecture per week (L – T): 3-1 Credits: 4

Course Outcomes (COs)

At the end of the course, the students will be able to:
1. **Create and Design** programs using a variety of data structures such as stacks, queues, hash tables, binary trees, heaps, graphs.
2. **Evaluate** and choose appropriate data structures to represent data items in real world.
3. **Analyze** the time and space complexities of algorithms.
4. **Implement and apply** sorting algorithms for problem solving.
5. **Understand** the concept of dynamic memory management.
6. **Identify and remember** user defined data types, linear data structures for solving real world problems.

Prerequisites

- Programming for Problem Solving

Detailed Content:

<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><strong>Introduction</strong>: Basic Terminologies: Elementary Data Organizations, Data Structure Operations: insertion, deletion, traversal etc.; Analysis of an Algorithm, Asymptotic Notations, Time-Space tradeoff. <strong>Searching</strong>: Linear Search and Binary Search techniques and their complexity analysis.</td>
<td>9</td>
</tr>
<tr>
<td>2</td>
<td><strong>Stacks and Queue</strong>: ADT Stack and its operations: Algorithms and their complexity analysis, Applications of Stacks: Expression Conversion and evaluation—Corresponding algorithms and complexity analysis. ADT queue, Types of Queue: Simple Queue, Circular Queue, Priority Queue; Operations on each type of Queues:</td>
<td>10</td>
</tr>
</tbody>
</table>
Algorithms and their analysis.

3 **Linked Lists**: Singly linked lists: Representation in memory, Algorithms of several operations: Traversing, Searching, Insertion into, Deletion from linked list; Linked representation of Stack and Queue, Header nodes, Doubly linked list: operations on it and algorithmic analysis; Circular Linked Lists: all operations their algorithms and the complexity analysis.

4 **Trees**: Basic Tree Terminologies, Different types of Trees: Binary Tree, Threaded Binary Tree, Binary Search Tree, AVL Tree; Tree operations on each of the trees and their algorithms with complexity analysis. Applications of Binary Trees. BTree, B+Tree: definitions, algorithms and analysis.

5 **Sorting and Hashing**: Objective and properties of different sorting algorithms: Selection Sort, Bubble Sort, Insertion Sort, Quick Sort, Merge Sort, Heap Sort; Performance and Comparison among all the methods, Hashing.

6 **Graph**: Basic Terminologies and Representations, Graph search and traversal algorithms and complexity analysis.

**Text book and Reference books:**
1) Data Structures and Program Design in C, 2/E by Robert L.Kruse, Bruce P.Leung.
3) Fundamentals of Data Structures of C by Ellis Horowitz, Sartaj Sahni, Susan Andersonfreed.
4) Data Structures in C by Aaron M. Tenenbaum.
5) Data Structures by S. Lipschutz.
6) Data Structures Using C by Reema Thareja.
9) Data Structures through C by Yashwant Kanetkar, BPB Publications.

**PCC-AIML-302: Computer Organization & Architecture**
**Contacts: 3L+1T per week**  **Credits: 4**

**Course Outcomes (COs):**

At the end of this course, students will demonstrate the ability to
1. Demonstrate how Computer Systems work & the basic principles
2. Understand Instruction Level Architecture and Instruction Execution
3. Understand the current state of art in memory system design
4. Demonstrate how I/O devices are accessed and its principles.
5. Impart the knowledge on micro programming control unit
6. Understand concepts of pipelining techniques.

**Prerequisites:**
1. Number Systems
2. Digital Electronics
Detailed Content:

<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Basic organization of the stored program computer and operation sequence for execution of a program. Role of operating systems and compiler/assembler. Fetch, decode and execute cycle, Concept of operator, operand, registers and storage. Commonly used number systems. Fixed and floating point representation of numbers. Overflow and underflow. Floating point - IEEE 754 standard.</td>
<td>9</td>
</tr>
<tr>
<td>2</td>
<td>Fixed point multiplication –Booth’s algorithm. Fixed point division - Restoring and non-restoring algorithms. Design of adders – serial adder, ripple carry and carry look-ahead principles. Design of ALU.</td>
<td>7</td>
</tr>
<tr>
<td>3</td>
<td>Instruction set architecture: Instruction format. Instruction length. 0-, 1-, 2-, 3-address instructions. Instruction cycle. Instruction sets and addressing modes. Introduction to RISC architectures. RISC vs CISC architectures.</td>
<td>7</td>
</tr>
<tr>
<td>4</td>
<td>Memory organization: Performance parameters, Memory unit design with special emphasis on implementation of CPU-memory interfacing. Memory hierarchy, hard disk, static and dynamic memory, associative memory. Cache memory, Virtual memory.</td>
<td>9</td>
</tr>
<tr>
<td>5</td>
<td>Design of control unit - hardwired and microprogrammed control. Introduction to instruction pipelining. Throughput and speedup, pipeline hazards.</td>
<td>6</td>
</tr>
<tr>
<td>6</td>
<td>I/O operations - Concept of handshaking, Polled I/O, interrupt and DMA.</td>
<td>4</td>
</tr>
</tbody>
</table>

Text book and Reference books:


BSC-AIML-301: Linear Algebra
Lecture per week (L – T): 3-0 Credits: 3

Course Outcomes (COs):

At the end of the course, the students will be able to:
1. **Formulate** elementary row and column operation.
2. **Evaluate** matrix algebra and related matrices to linear transformation.
3. **Analyze** singular value decomposition.
4. **Solve** systems of linear equations.
5. Use matrix algebra and the related matrices to linear transformations.
6. **Understand** the basic ideas of linear mapping.

Prerequisites:
Mathematics-I
Mathematics-II

Detailed Content:

<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Systems of linear equations, Matrices, Elementary row operations, Row-reduced echelon matrices. Vector spaces, Subspaces, Bases and dimension, Ordered bases and coordinates.</td>
<td>9</td>
</tr>
<tr>
<td>2</td>
<td>Linear transformations, Rank-nullity theorem, Algebra of linear transformations, Isomorphism, Matrix representation, Linear functionals, Annihilator, Double dual, Transpose of a linear transformation.</td>
<td>10</td>
</tr>
<tr>
<td>3</td>
<td>Characteristic values and characteristic vectors of linear transformations, Diagonalizability, Minimal polynomial of a linear transformation, Cayley-Hamilton theorem, Invariant subspaces, Direct-sum decompositions, Invariant direct sums, The primary decomposition theorem, Cyclic subspaces and annihilators, Cyclic decomposition, Rational, Jordan forms.</td>
<td>10</td>
</tr>
<tr>
<td>4</td>
<td>Inner product spaces, Orthonormal bases, Gram-Schmidt process.</td>
<td>7</td>
</tr>
</tbody>
</table>

Text books and Reference books:

1. E Kreyszig, Advanced Engineering Mathematics, Wiley-India
10. Anton and Rorres, Elementary Linear Algebra, Applications version, Wiley India Edition

HSMC-AIML-301: Economics for Engineers
Contacts: 2L per week Credits: 2

Course Outcomes (COs):

At the end of the course, the students will be able to:
- Evaluate the economic theories, cost concepts and pricing policies.
- Understand the market structures and integration concepts.
- Understand the measures of national income, the functions of banks and concepts of globalization.
- Apply the concepts of financial management for project appraisal.
- Understand accounting systems and analyze financial statements using ratio analysis.
- Understand the impact of inflation, taxation, depreciation. Financial planning, economic basis for replacement, project scheduling, and legal and regulatory issues are introduced and applied to economic investment and project-management problems.
**Detailed Content:**

<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
</table>
4. Cash Flow & Rate of Return Analysis – Calculations, Treatment of Salvage Value, Annual Cash Flow Analysis, Analysis Periods; Internal Rate of Return, Calculating Rate of Return, Incremental Analysis; Best Alternative Choosing an Analysis Method, Future Worth Analysis, Benefit-Cost Ratio Analysis, Sensitivity and Breakeven Analysis. Economic Analysis In The Public Sector -Quantifying And Valuing Benefits & drawbacks. | 9 |
| 3      | 5. Inflation and Price Change – Definition, Effects, Causes, Price Change with Indexes, Types of Index, Composite vs Commodity Indexes, Use of Price Indexes In Engineering Economic Analysis, Cash Flows that inflate at different Rates.  

**Text books and Reference books:**

2. Donald Newnan, Ted Eschembach, Jerome Lavelle: Engineering Economics Analysis, OUP  
5. R.Paneer Seelvan: Engineering Economics, PHI  
ESC-AIML-391: Digital Electronics Lab
Contacts: 3P per week Credits: 1.5

Course Outcomes (COs):

At the end of this course, students will demonstrate the ability to
1. Acquire knowledge about basics of digital electronics.
2. Explain about how to solve problems related to number systems and Boolean algebra.
3. Identify, analyze and design combinational circuits.
4. Design BCD to Excess-3 and Binary to gray code conversion circuit.
5. Compare various synchronous and asynchronous sequential circuits.
6. Analyze sequential digital circuits like flip-flops, registers, counters.

<table>
<thead>
<tr>
<th>Laboratory Experiments:</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
</tr>
<tr>
<td>2</td>
</tr>
<tr>
<td>3</td>
</tr>
<tr>
<td>4</td>
</tr>
<tr>
<td>5</td>
</tr>
<tr>
<td>6</td>
</tr>
<tr>
<td>7</td>
</tr>
<tr>
<td>8</td>
</tr>
<tr>
<td>9</td>
</tr>
<tr>
<td>10</td>
</tr>
<tr>
<td>11</td>
</tr>
</tbody>
</table>

PCC-AIML-391: Data Structure and Algorithms Lab
Labs per week (P): 3 Credits: 1.5

Course Outcomes (COs):

At the end of the course, the students will be able to:
1. Design and solve programs using a variety of data structures such as stacks, queues, hash tables, binary trees, heaps, graphs.
2. Evaluate and choose appropriate data structures to represent data items in real world.
3. Analyze the time and space complexities of algorithms.
4. Implement sorting and searching algorithms for problem solving.
5. Understand the concept of dynamic memory management.
6. Identify and remember user defined data types, linear data structures for solving real world problems.

Prerequisites:
- Programming for Problem Solving
- Data Structure and Algorithms
Detailed Content:

1. Implementation of array operations
2. Stacks and Queues: adding, deleting elements Circular Queue: Adding & deleting elements
3. Application of Stack: Expression Evaluation, Expression Conversion
4. Implementation of linked lists: inserting, deleting, and inverting a linked list. Implementation of stacks & queues using linked lists
5. Polynomial addition, Polynomial multiplication
6. Recursive and Non-recursive traversal of Trees
7. Threaded binary tree traversal. AVL tree implementation
8. Implementation of different searching & sorting techniques.

Text book and Reference books:

1) Data Structures and Program Design in C, 2/E by Robert L.Kruse, Bruce P.Leung.
3) Fundamentals of Data Structures of C by Ellis Horowitz, Sartaj Sahni, Susan Andersonfreed.
4) Data Structures in C by Aaron M. Tenenbaum.
5) Data Structures by S. Lipschutz.
6) Data Structures Using C by Reema Thareja.
9) Data Structures through C by Yashwant Kanetkar, BPB Publications.

PCC-AIML-392: Computer Architecture Lab
Contacts: 3P per week Credits: 1.5

Course Outcomes:
At the end of this course, students will demonstrate the ability to
- **Summarize** Xilinx/ Altera (VHDL) foundation tools and Hardware Description Language.
- **Demonstrate** different concepts and methods of digital system design techniques through hands-on projects.
- **Build** various combinational and sequential digital systems.
- **Identify** knowledge, techniques required to design, implement and test modern day digital systems.
- **Evaluate and interpret** the results of logic and timing simulations.
- **Analyze** digital systems through hands-on experiments on the Xilinx/ Altera tools.

Laboratory Experiments:

1. Write VHDL codes for various logic gates.
2. Using VHDL, design a half adder in data flow model.
3. Using VHDL, design a full adder in data flow model.
4. Using VHDL, design a half subtractor.
5. Using VHDL, design a full subtractor.
6. Using VHDL, design 1-bit comparator.
7. Using VHDL, design 4:1 Multiplexer in data flow model.
8. Using VHDL, design 2:4 Decoder in data flow model.
Using VHDL, design 1:4 DEMUX in data flow model.
10 Write VHDL code for basic gates: 2 i/p AND Gate (Multiple Bit Handling)
11 Using VHDL, design 4:1 Multiplexer using when-else structure.
12 Using VHDL, design 2: 4 Decoder using when-case structure.
13 Using VHDL, design 2: 4 Decoder in behavioral model.
14 Write VHDL code for 4-bit Up Counter
15 Write VHDL code for 4-bit Down Counter
16 Write VHDL code for 4-bit Up-Down Counter
17 Using VHDL, design SR-flip flop in behavioral model.
18 Write VHDL code for D Flip Flop
19 Using VHDL, design JK-flip flop in behavioral model.
20 Write VHDL code for T Flip Flop

PCC-AI ML-393: IT Workshop (Python)
Labs per week: 3 Credits: 1.5

Course Outcomes (COs):

At the end of the course, the students will be able to:
1. **Develop** algorithmic solutions to simple computational problems.
2. **Identify** and repair coding errors in a program.
3. **Demonstrate** programs using simple Python statements and expressions.
4. **Explain** control flow and functions concept in Python for solving problems.
5. **Use** Python data structures – lists, tuples & dictionaries for representing compound data.
6. **Explain** files, exception, modules and packages in Python for solving problems.

Prerequisites:
- Programming for Problem Solving

Detailed Content:

<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><strong>Introduction</strong>: History, Features, Setting up path, Working with Python, Basic Syntax, Variable and Data Types, Operator</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td><strong>Conditional Statements</strong>: If, If- else, Nested if-else, Looping, For, While, Nested loops</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td><strong>Control Statements</strong>: Break, Continue, Pass</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td><strong>String Manipulation</strong>: Accessing Strings, Basic Operations, String slices, Function and Methods</td>
<td>3</td>
</tr>
<tr>
<td>5</td>
<td><strong>Lists</strong>: Introduction, Accessing list, Operations, Working with lists, Function and Methods</td>
<td>3</td>
</tr>
<tr>
<td>6</td>
<td><strong>Tuple</strong>: Introduction, Accessing tuples, Operations, Working, Functions and Methods</td>
<td>3</td>
</tr>
<tr>
<td>7</td>
<td><strong>Dictionaries</strong>: Introduction, Accessing values in dictionaries, Working with dictionaries, Properties</td>
<td>3</td>
</tr>
<tr>
<td>8</td>
<td><strong>Functions</strong>: Defining a function, Calling a function, Types of functions, Function Arguments, Anonymous functions, Global and local variables</td>
<td>3</td>
</tr>
<tr>
<td>9</td>
<td><strong>Modules</strong>: Importing module, Math module, Random module, Packages, Composition, Input-Output Printing on screen, Reading data from keyboard,</td>
<td>3</td>
</tr>
</tbody>
</table>
Exception Handling: Exception, Exception Handling, Except clause, Try abd finally clause, User Defined Exceptions.

Laboratory Experiments:
1. Practical Assignments related with implementation of PCC-AIML-393

Text book and Reference books:
5. TanejaSheetal and Kumar Naveen, “Python Programming |A modular approach”, Pearson Education,

Semester - IV
ESC-AIML-401: Probability and Statistics
Lecture per week (L – T): 3-0   Credits: 3

Course Outcomes (COs):

At the end of the course, the students will be able to:
1. Understand basic notions of probability arising in a variety of uncertain situations which are nontraditional in areas of science and engineering.
2. Understand the concepts of Binomial and Poisson distributions – Normal distribution, Exponential distributions, Weibull distribution.
3. Analyze the Random variables, one dimensional Random Variables, Discrete and Continuous RV-Density and Distribution function of RV.
4. Discuss the basic ideas of Statistics.
5. Explain the ideas of Correlation and regression.
6. Illustrate the concepts on Test of Significance, Test of Hypothesis.

Prerequisites:

- Basic Mathematics.

Detailed Content:

<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Probability Theory: Introduction to probability concepts, Random experiments, Events, Conditional probability, Independent events, Theorem of Total Probability, Baye’s theorem Sample space</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>Random variables (RV):</td>
<td>12</td>
</tr>
</tbody>
</table>

Special Distributions:
- Binomial and Poisson distributions – Normal distribution, Exponential distributions, Weibull distribution

| 3 | Correlation and regression:
|   | Two dimensional random variables, Joint distribution and Joint density functions - Marginal, Conditional Distribution and Density functions. Regression and Correlation. – Partial and Multiple Correlation- Multiple Regression. |

| 4 | Test of Significance:
|   | Testing of hypothesis – Introduction - Types of errors, critical region, procedure of testing hypothesis.
|   | Large sample tests - Z test for Single Proportion, Difference of Proportion, Single mean and difference of means. Small sample tests - Student’s t-test, F-test - Chi-square test - Goodness of fit - Independence of Attributes

| 5 | Design of Experiments:
|   | Analysis of variance – One and Two way classifications – Completely Randomized Design (CRD), Randomized Block Design (RBD), Latin Square Design (LSD)

Text books and Reference books:

4. Fundamental of Mathematical Statistics by T Veerarajan, Yes Dee Publishing Pvt Ltd.

PCC-AIML-401: Object Oriented Programming & Java
Lecture per week (L – T): 3-0 Credits: 3

Course Outcomes (COs):

At the end of the course, the students will be able to:

1. Create and explain GUI and thread based application.
2. Evaluate the complexity of procedural language by using the concept of polymorphism, inheritance, abstraction, and encapsulation.
3. Analyze any real world problem with object oriented approach and formulate a solution for the same.
4. Implement and apply object oriented approach to relate to real world problem.
5. Understand, describe object oriented programming.
6. Recall the knowledge of procedural language and map it to paradigm of Object Oriented concept.

Prerequisites:
- Programming for Problem Solving

Detailed Content:
<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><strong>Basics of OOP and Introduction to JAVA:</strong> Properties of object oriented programming language, Object, Class, relationships among objects. Aggregation, Association, Generalization, meta-class. Class, object, message passing, inheritance, encapsulation, polymorphism. Basic concept of JAVA programming—advantages of java, byte-code &amp; JVM, data types, operators, control statements &amp; loops, array, creation of class, object, constructor, finalize and garbage collection.</td>
<td>8</td>
</tr>
<tr>
<td>2</td>
<td><strong>Class &amp; Object proprieties:</strong> Different types of access specifiers, method overloading, this keyword, use of objects as parameter &amp; methods returning objects, call by value &amp; call by reference, static variables, methods and block nested &amp; inner classes. Inbuilt classes like String, Character, StringBuffer, basic string handling concepts, concept of mutable and immutable string. <strong>Reusability properties:</strong> Super class &amp; subclasses including multilevel hierarchy, process of constructor calling in inheritance, use of super and final keywords with super() method, dynamic method dispatch, abstract classes &amp; methods, interfaces. Creation of packages, importing packages, member access for packages, UTIL package.</td>
<td>10</td>
</tr>
<tr>
<td>3</td>
<td><strong>Exception handling and I/O:</strong> Exception handling basics, different types of exception classes, use of try &amp; catch with throw, throws &amp; finally, creation of user defined exception classes. Input Output stream structure, Wrapper class, command line arguments, basics of I/O operations – keyboard input using Buffered Reader&amp; Scanner classes. File copy programming using command line arguments.</td>
<td>6</td>
</tr>
<tr>
<td>4</td>
<td><strong>Multithreading and Applet &amp; Swing Programming:</strong> Basics of multithreading, main thread, thread life cycle, creation of multiple threads, thread priorities, thread synchronization, inter-thread communication, deadlocks for threads, suspending &amp; resuming threads. Introduction to Microservices. Basics of applet programming, applet life cycle, difference between application &amp; applet programming, parameter passing in applets. Basic of swing programming, Difference between applet and swing,</td>
<td>7</td>
</tr>
</tbody>
</table>

**Text book and Reference books:**

1) Object Oriented Modelling and Design, Rambaugh, James Michael, Blaha, Prentice Hall, India.
2) Object Oriented System Development Ali Bahrami,McGraw Hill.
3) The complete reference-Java2, Patrick Naughton, Herbert Schildt, TMH.
4) Core Java For Beginners, R.KDas,VIKAS PUBLISHING.
5) Java How to Program, Deitel and Deitel, 6th Ed. – Pearson.
6) Beginning Java 2 SDK, IvorHorton’s, Wrox.
7) Programming With Java: A Primer, E. Balagurusamy, 3rd Ed., TMH.

**PCC-AIML-402: Formal Language & Automata Theory**

Contacts: **3L**  Credit: **3**

**Course Outcomes (COs):**

After completion of the course, students will be able to:

1. **Design** finite automata to accept a set of strings of a language.
2. For a given language **evaluate** whether the given language is regular or not.
3. **Develop** context free grammars to generate strings of context free language.
4. Determine equivalence of languages accepted by Push down Automata and languages generated by context free grammars.
5. Implement the hierarchy of formal languages, grammars and machines.
6. Distinguish between computability and non-computability and Decidability and undecidability.

Pre-Requisite:
NIL

Detailed Content:

<table>
<thead>
<tr>
<th>Unit</th>
<th>Content</th>
<th>Hrs/Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Introduction: Alphabet, languages and grammars, productions and derivation, Chomsky hierarchy of languages.</td>
<td>6</td>
</tr>
<tr>
<td>2</td>
<td>Regular languages and finite automata: Regular expressions and languages, deterministic finite automata (DFA) and equivalence with regular expressions, nondeterministic finite automata(NFA) and equivalence with DFA, regular grammars and equivalence with finite automata, properties of regular languages, pumping lemma for regular languages, minimization of finite automata</td>
<td>7</td>
</tr>
<tr>
<td></td>
<td>Context-free languages and pushdown automata: Context-free grammars (CFG) and languages (CFL),Chomsky and Greibach normal forms, nondeterministic pushdown automata (PDA) and equivalence with CFG, parse trees, ambiguity in CFG, pumping lemma for context-free languages, deterministic push down automata, closure properties of CFLs.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Context-sensitive languages: Context-sensitive grammars (CSG) and languages, linear bounded automata and equivalence with CSG.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Turing machines: The basic model for Turing machines (TM), Turing recognizable(recursively enumerable) and Turing-decidable (recursive)languages and their closure properties, variants of Turing machines, nondeterministic TMs and equivalence with deterministic TMs, unrestricted grammars and equivalence with Turing machines, TMs as enumerators</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Undecidability: Church-Turing thesis, universal Turing machine, the universal and diagonalization languages, reduction between languages and Rice's theorem, undecidable problems about languages</td>
<td></td>
</tr>
</tbody>
</table>

Text books and reference books:

PCC-AIML-403: Design and Analysis of Algorithm
Contacts: 3L + 1T per week    Credits: 4
Course Outcomes (COs):

At the end of the course, the students will be able to:

1. **Prove** the correctness and analyze the running time of the basic algorithms for those classic problems in various domains.
2. **Apply** the algorithms and design techniques to solve problems.
3. **Analyze** the complexities of various problems in different domains.
4. **Apply** design and development principles in the construction of software systems of varying complexity.
5. **Design** and development principles in the construction of software systems of varying complexity.
6. **Adapt** current techniques, skills, and tools necessary for computing practice.

Prerequisites:
- Basics of C programming
- Data Structure and Algorithms

Detailed Content:

<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Models of computation: RAM, TM etc. time and space complexity&lt;br&gt;Asymptotic Notation Big-O, omega, theta etc.; finding time complexity of well-known algorithms like- heapsort, search algorithm etc.&lt;br&gt;Algorithm Design techniques&lt;br&gt;Recursion- Definition, Use, Limitations, Examples: Hanoi problem, Tail Recursion</td>
<td>7</td>
</tr>
<tr>
<td>2</td>
<td>Divide and Conquer&lt;br&gt;Basic method, use, Examples: Merge sort, Quick Sort, Binary Search,&lt;br&gt;Dynamic Programming&lt;br&gt;Basic method, use, Examples: matrix-chain multiplication, All pair shortest paths, single-source shortest path, Travelling Salesman problem</td>
<td>8</td>
</tr>
<tr>
<td>3</td>
<td>Branch and Bound: Basic method, use, Examples: The 15-puzzle problem&lt;br&gt;Backtracking: Basic method, use, Examples: Eight queens problem, Graph coloring problem, Hamiltonian problem</td>
<td>6</td>
</tr>
<tr>
<td>4</td>
<td>Greedy Method: Basic method, use, Examples: Knapsack problem, Job sequencing with deadlines, minimum spanning tree (Prim's and Kruskal's algorithms)&lt;br&gt;Lower Bound Theory: Bounds on sorting and sorting techniques using partial and total orders.&lt;br&gt;Disjoint Set Manipulation: Set manipulation algorithm like UNION-FIND, union by rank, Path compression</td>
<td>7</td>
</tr>
<tr>
<td>5</td>
<td>Properties of graphs and graph traversal algorithms: BFS and DFS&lt;br&gt;Matrix manipulation algorithms: Different types of algorithms and solution of simultaneous equations, DFT &amp; FFT algorithm; integer multiplication schemes</td>
<td>6</td>
</tr>
<tr>
<td>6</td>
<td>Notion of NP-completeness: P class, NP-hard class, NP-complete class, Circuit Satisfiability problem, Clique Decision Problem.&lt;br&gt;Approximation algorithms: Necessity of approximation scheme, performance guarantee, Polynomial time</td>
<td>6</td>
</tr>
</tbody>
</table>
approximation schemes: 0/1 knapsack problem.

Text book and Reference books:

1. A. Aho, J. Hopcroft and J. Ullman “The Design and Analysis of algorithms”
4. Goodman: Introduction to Design and Analysis of Algorithms, TMH.
6. S. Baase “Computer algorithms”
8. A. Borodin and I. Munro, “The computational complexity of Algebraic and Numeric problems”

ESC-AIML-402: Signals and Systems
Contacts: 3L per week Credits: 3

Course Outcomes:

At the end of this course, students will demonstrate the ability to

- Analyze different types of signals
- Represent continuous and discrete systems in time and frequency domain using different transforms
- Analyse systems in complex frequency domain.
- Understand sampling theorem and its implications.
- Investigate whether the system is stable
- Sampling and reconstruction of a signal

Prerequisites:
- Linear Algebra

Detailed Content:

<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Signals and systems as seen in everyday life, and in various branches of engineering and science. Energy and power signals, continuous and discrete time signals, continuous and discrete amplitude signals. System properties: linearity: additivity and homogeneity, shift-invariance, causality, stability, realizability.</td>
</tr>
<tr>
<td>3</td>
<td>Periodic and semi-periodic inputs to an LSI system, the notion of a frequency response and its relation to the impulse response, Fourier series representation, the Fourier Transform, convolution/multiplication and their effect in the frequency domain, magnitude and phase response, Fourier domain duality. The Discrete-Time Fourier Transform (DTFT) and the Discrete Fourier Transform (DFT). Parseval's Theorem. The idea of signal space and orthogonal bases</td>
</tr>
<tr>
<td></td>
<td>Hours/ Module</td>
</tr>
<tr>
<td>1</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>6</td>
</tr>
<tr>
<td>3</td>
<td>7</td>
</tr>
<tr>
<td>4</td>
<td>The Laplace Transform, notion of eigen functions of LSI systems, a basis of eigen functions, region of convergence, poles and zeros of system, Laplace domain analysis, solution to differential equations and system behavior. The z-Transform for discrete time signals and systems- eigen functions, region of convergence, z-domain analysis.</td>
</tr>
<tr>
<td>---</td>
<td>---</td>
</tr>
</tbody>
</table>


Text book and Reference books:


MC-AIML-401: Environmental Sciences

Contacts: 2L per week    Credits: 0

Course Outcomes (COs):

At the end of this course, students will demonstrate the ability to

- Acquire fundamental knowledge of different aspects of environment and local, regional and global environmental problems.
- Get the information about ecosystem and also about its functions like Food chain, Ecological pyramids etc.
- Obtain the knowledge about the different types of resources like land, water, mineral and energy and also about the effects of environment by the usage of these resources.
- Gain the knowledge about the ecosystem diversity, its values and also about the importance of the endemic species and different techniques involved in its conservation.
- Gain the knowledge about the different types of pollutions and their control technologies, Waste water treatment, Bio medical waste management etc.,
- Acquire the complete information about EIA- Environmental Impact Assessment, Sustainable developmental activities, environmental policies and regulations, awareness among people about protection of wild life, forest and other natural resources.

Prerequisites:

- Basics of Chemistry

Detailed Content:
<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><strong>Basic ideas of environment, basic concepts, man, society &amp; environment, their interrelationship</strong> (1L) Mathematics of population growth and associated problems, Importance of population study in environmental engineering, definition of resource, types of resource, renewable, non-renewable, potentially renewable, effect of excessive use vis-à-vis population growth, Sustainable Development. (2L) Materials balance: Steady state conservation system, steady state system with non-conservative pollutants, step function. (1L) <strong>Environmental degradation:</strong> Natural environmental Hazards like Flood, earthquake, Landslide-causes, effects and control/management; Anthropogenic degradation like Acid rain-cause, effects and control. Nature and scope of Environmental Science and Engineering. (2L)</td>
<td>6</td>
</tr>
<tr>
<td>2</td>
<td><strong>Elements of ecology:</strong> System, open system, closed system, definition of ecology, species, population, community, definition of ecosystem- components types and function. (1L) <strong>Structure and function of the following ecosystem:</strong> Forest ecosystem, Grassland ecosystem, Desert ecosystem, Aquatic ecosystems, Mangrove ecosystem (special reference to Sundarban); Food chain [definition and one example of each food chain], Food web. (2L) <strong>Biogeochemical Cycle:</strong> definition, significance, flow chart of different cycles with only elementary reaction [Oxygen, carbon, Nitrogen, Phosphate, Sulphur]. (1L) <strong>Biodiversity:</strong> types, importance, Endemic species, Biodiversity Hot-spot, Threats to biodiversity, Conservation of biodiversity.</td>
<td>6</td>
</tr>
<tr>
<td>3</td>
<td><strong>Atmospheric Composition:</strong> Troposphere, Stratosphere, Mesosphere, Thermosphere, Tropopause and Mesopause. (1L) <strong>Energy balance:</strong> Conductive and Convective heat transfer, radiation heat transfer, simple global temperature model [Earth as a black body, earth as albedo], Problems. (1L) <strong>Green house effects:</strong> Definition, impact of greenhouse gases on the global climate and consequently on sea water level, agriculture and marine food. Global warming and its consequence, Control of Global warming. Earth’s heat budget. (1L) <strong>Lapse rate:</strong> Ambient lapse rate, Adiabatic lapse rate, atmospheric stability, temperature inversion (radiation inversion). (2L) <strong>Atmospheric dispersion:</strong> Maximum mixing depth, ventilation coefficient, effective stack height, smokestack plumes and Gaussian plume model. (2L) <strong>Definition of pollutants and contaminants:</strong> Primary and secondary pollutants: emission standard, criteria pollutant. Sources and effect of different air pollutants Suspended particulate matter, oxides of carbon, oxides of nitrogen, oxides of sulphur, particulate, PAN. (2L) <strong>Smog, Photochemical smog and London smog:</strong> Depletion Ozone layer: CFC, destruction of ozone layer by CFC, impact of other greenhouse gases, effect of ozone modification. (1L) <strong>Standards and control measures:</strong> Industrial, commercial and residential air quality standard, control measure (ESP, cyclone separator, bag house, catalytic converter, scrubber (ventury), Statement with brief reference).</td>
<td>11</td>
</tr>
<tr>
<td>4</td>
<td><strong>Hydrosphere, Hydrological cycle and Natural water.</strong> Pollutants of water, their origin and effects: Oxygen demanding wastes, pathogens, nutrients, Salts, thermal application, heavy metals, pesticides, volatile organic compounds. (2L) <strong>River/Lake/ground water pollution:</strong> River: DO, 5-day BOD test, Seeded BOD test, BOD reaction rate constants, Effect of oxygen demanding wastes on river [deoxygenation, reaeration], COD, Oil, Greases, pH. (2L) <strong>Lake:</strong> Eutrophication [Definition, source and effect]. (1L) <strong>Ground water:</strong> Aquifers, hydraulic gradient, ground water flow (Definition only) (1L) <strong>Standard and control:</strong> Waste water standard [BOD, COD, Oil, Grease], Water Treatment system [coagulation and flocculation, sedimentation and filtration, disinfection, hardness and alkalinity, softening] Waste water treatment system, primary and secondary treatments [Trickling filters, rotating biological contractor, Activated sludge, sludge treatment,</td>
<td>9</td>
</tr>
</tbody>
</table>
oxidation ponds] tertiary treatment definition. (2L) Water pollution due to the toxic elements and their biochemical effects: Lead, Mercury, Cadmium, and Arsenic (1L)

5 Lithosphere; Internal structure of earth, rock and soil (1L) Solid Waste: Municipal, industrial, commercial, agricultural, domestic, pathological and hazardous solid wastes; Recovery and disposal method- Open dumping, Land filling, incineration, composting, recycling. Solid waste management and control (hazardous and biomedical waste).(2L) 3

6 Definition of noise, effect of noise pollution, noise classification [Transport noise, occupational noise, neighbourhood noise] (1L) Definition of noise frequency, noise pressure, noise intensity, noise threshold limit value, equivalent noise level, L10 (18hr Index),n Ld. Noise pollution control. (1L) 3

7 Environmental impact assessment, Environmental Audit, Environmental laws and protection act of India, Different international environmental treaty/ agreement/ protocol. (2L) 2

Text book and Reference books:

PCC-AIML-491: Object Oriented Programming & Java Lab
Lab per week (P): 3  Credits: 1.5

Course Outcomes (COs):

At the end of the course, the students will be able to:
1. **Create** multithreaded programs.
2. **Evaluate** the complexity of procedural language by using the concept of polymorphism, inheritance, abstraction, and encapsulation.
3. **Experiment** any real world problem with object oriented approach and formulate a solution for the same.
4. **Implement and apply** object oriented approach to relate to real world problem.
5. **Understand** and develop graphical user interface using AWT.
6. **Recall** the knowledge of event handling mechanism.

Prerequisites:
- Programming for Problem Solving
- Object Oriented Programming & Java

Detailed Content:

- Implement all problems abiding by features of object oriented programming (Abstraction, Encapsulation, Reusability, Data Hiding, Generalization, and Specialization.)
- Familiarization on object oriented approach of programming: use of class, object, and reference.
- Use of constructor, static, final, array, date, access specifiers.
- Familiarization with String, StringBuffer, ArrayList and LinkedList classes.
- Familiarization on Inheritance and Dynamic Method Dispatch.
- Familiarization on Abstract Class, Interface and Package Java Exception Handling.
- Familiarization on Java IO using Scanner, BufferedReader, PrintWriter. File handling in Java.
- Exploring Java multithreading concept.
- Familiarization on Java Applet, AWT Event Handling.
- Basics of Java Swing: Different Layouts, Event Handling.

Text books and Reference books:

1) Object Oriented Modelling and Design, Rambaugh, James Michael, Blaha, Prentice Hall, India.
2) Object Oriented System Development Ali Bahrami, McGraw Hill.
3) The complete reference-Java2, Patrick Naughton, Herbert Schildt, TMH.
4) Core Java For Beginners, R.KDas, VIKAS PUBLISHING.
5) Java How to Program, Deitel and Deitel, 6th Ed. – Pearson.
6) Beginning Java 2 SDK, IvorHorton's, Wrox.
7) Programming With Java: A Primer,E. Balagurusamy,3rd Ed., TMH.

PCC-AIML-492: Design and Analysis of Algorithm Lab
Lab per week (P): 3 Credits: 1.5

Course Outcomes (COs):

After completion of this course, the students are able to:

1) **Solve** problems by applying appropriate algorithms.
2) **Analyze** the efficiency of various algorithms.
3) **Apply** techniques of stacks and queues to solve problems.
4) **Develop** a program that can be solved in many ways using different techniques.
5) **Identify** and evaluate complex problems using principles of mathematics and engineering science
6) **Design** a novel solution for real life problem

Prerequisites:
- Basics of C programming
- Data Structure and Algorithms

Lab Experiments List:

1. Design, develop and implement the specified algorithms for the following problems using C/C++ Language in LINUX environment.
2. Write a C/C++ program to sort the elements by using quick sort method.
3. Write a C/C++ program to sort the elements by using merge sort method.
4. Obtain the Topological ordering of vertices in a given digraph.
5. From a given vertex in a weighted connected graph, find shortest paths to other vertices using Dijkstra’s algorithm.
6. Implement 0/1 Knapsack problem using Dynamic Programming.
7. Find Minimum Cost Spanning Tree of a given undirected graph using Kruskal’s algorithm.
8. Find Minimum Cost Spanning Tree of a given undirected graph using Prim’s algorithm.
9. Compute the transitive closure of a given directed graph using Warshall’s algorithm.
10. Implement All-Pairs Shortest Paths Problem using Floyd’s algorithm.
11. Print all the nodes reachable from a given starting node in a digraph using BFS method.
12. Check whether a given graph is connected or not using DFS method.
13. Implement N Queen’s problem using Back Tracking
Text/ Reference Books:

E-Resources:

Semester - V

BSC-AIML-501: Discrete Mathematics
Lecture per week (L – T): 3-0 Credits: 3

Course Outcomes (COs):

At the end of the course, the students will be able to:
1. Apply mathematical logic to solve problems.
2. Understand sets, relations, functions and discrete structures.
3. Use logical notations to define and reason about fundamental mathematical concepts such as sets relations and functions.
4. Identify functions and determine their properties.
5. Formulate problems and solve recurrence relations.
6. Model and solve real world problems using graphs and trees.

Prerequisites:
1. Basic Mathematics
2. Probability and Statistics

Detailed Content:

<table>
<thead>
<tr>
<th>Module No</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Introduction: Objective, scope and outcome of the course.</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>Propositional Logic: Proposition, First order logic, Basic logical operation, truth tables, tautologies, Contradictions, Algebra of Proposition, logical implications,</td>
<td>6</td>
</tr>
<tr>
<td>5</td>
<td>Algebraic Structures: Definition, Properties, types: Semi Groups, Monoid, Groups, Abelian group, properties of groups, Subgroup, cyclic groups, Cosets, factor group, Permutation groups, Normal subgroup, Homomorphism and isomorphism of Groups, example and standard results, Rings and Fields: definition and standard results.</td>
<td>5</td>
</tr>
<tr>
<td>6</td>
<td>Graph Theory: Introduction and basic terminology of graphs, Planer graphs, Multigraphs and weighted graphs, Isomorphic graphs, Paths, Cycles and connectivity, Shortest path in weighted graph, Introduction to Eulerian paths and circuits, Hamiltonian paths and circuits, Graph colouring, chromatic number, Isomorphism and Homomorphism of graphs, matching, vertex/edge covering.</td>
<td>6</td>
</tr>
<tr>
<td>Total</td>
<td></td>
<td>32</td>
</tr>
</tbody>
</table>

Text/Reference Books:

1. Discrete Mathematics with Applications, Koshy, Elsevier
2. Discrete Mathematical Structures By Lipshutz & Lipson, TMH
3. Discrete Mathematical Structures, Kolman et al, Pearson

PCC-AIML-501: Operating System
Contacts: 3L per week Credits: 3

Course Outcomes:
At the end of this course, students will demonstrate the ability to
- Understand the mechanisms of OS to handle processes and threads and their communication
- Develop algorithms for process scheduling for a given specification of CPU utilization, Throughput, Turnaround Time, Waiting Time, Response Time.
- Demonstrate the mechanisms involved in memory management in contemporary OS
- Understand the components and management aspects of concurrency management
- Develop the techniques for optimally allocating memory to processes by increasing memory utilization and for improving the access time.
- Design and implement file management system.

Prerequisites:
- Digital Electronics
- Computer Organization & Architecture
- Programming Concepts
## Detailed Content:

<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>Inter-process Communication: Critical Section, Race Conditions, Mutual Exclusion, Hardware Solution, Strict Alternation, Peterson’s Solution, The Producer-Consumer Problem, Semaphores, Event Counters, Monitors, Message Passing, Classical IPC Problems: Reader’s &amp; Writer Problem, Dinning Philosopher Problem etc.</td>
<td>6</td>
</tr>
<tr>
<td>4</td>
<td>Deadlocks: Definition, Necessary and sufficient conditions for Deadlock, Deadlock Prevention, Deadlock Avoidance: Banker’s algorithm, Deadlock detection and Recovery.</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>Memory Management: Basic concept, Logical and Physical address map, Memory allocation: Contiguous Memory allocation – Fixed and variable partition– Internal and External fragmentation and Compaction; Paging: Principle of operation – Page allocation – Hardware support for paging, Protection and sharing, Disadvantages of paging, Virtual Memory: Basics of Virtual Memory – Hardware and control structures – Locality of reference, Page fault , Working Set , Dirty page/Dirty bit – Demand paging, Page Replacement algorithms: Optimal, First in First Out (FIFO), Second Chance (SC), Not recently used (NRU) and Least Recently used (LRU).</td>
<td>8</td>
</tr>
</tbody>
</table>

### Text book and Reference books:
PCC-AIML-502: Database Management Systems
Contacts: 3L per week Credits: 3

Course Outcomes (COs):

On completion of the course students will be able to:

1. Describe the fundamental elements of relational database management systems
2. Explain the basic concepts of relational data model, entity-relationship model, relational database design, relational algebra and SQL.
3. Design ER-models to represent simple database application scenarios
4. Convert the ER-model to relational tables, populate relational database and formulate SQL queries on data.
5. Improve the database design by normalization.
6. Understand the basic database storage structures and access techniques: file and page organizations, indexing methods including B tree, and hashing.

Prerequisites:
NIL

Detailed Content:

<table>
<thead>
<tr>
<th>Unit</th>
<th>Content</th>
<th>Hrs/Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><strong>Database system architecture:</strong> Data Abstraction, Data Independence, Data Definition Language (DDL), Data Manipulation Language (DML). Data models: Entity-relationship model, network model, relational and object oriented data models, integrity constraints, data manipulation operations.</td>
<td>9</td>
</tr>
<tr>
<td>2</td>
<td><strong>Relational query languages:</strong> Relational algebra, Tuple and domain relational calculus, SQL3, DDL and DML constructs, Open source and Commercial DBMS - MYSQL, ORACLE, DB2, SQL server. Relational database design: Domain and data dependency, Armstrong’s axioms, Normal forms, Dependency preservation, Lossless design. Query processing and optimization: Evaluation of relational algebra expressions, Query equivalence, Join strategies, Query optimization algorithms.</td>
<td>13</td>
</tr>
<tr>
<td>3</td>
<td><strong>Storage strategies:</strong> Indices, B-trees, hashing.</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td><strong>Transaction processing:</strong> Concurrency control, ACID property, Serializability of scheduling, Locking and timestamp based schedulers, Multi-version and optimistic Concurrency Control schemes, Database recovery.</td>
<td>5</td>
</tr>
<tr>
<td>5</td>
<td><strong>Database Security:</strong> Authentication, Authorization and access control, DAC, MAC and RBAC models, Intrusion detection, SQL injection.</td>
<td>3</td>
</tr>
<tr>
<td>6</td>
<td><strong>Advanced topics:</strong> Object oriented and object relational databases, Logical databases, Web databases, Distributed databases, Data warehousing and data mining.</td>
<td>3</td>
</tr>
</tbody>
</table>
Text book and Reference books:


PCC-AIML-503: Machine Learning Foundation
Lecture per week: 3 Credits: 3

Course Outcomes (COs):

At the end of the course, the students will be able to:

1. Develop an appreciation for what is involved in learning models from data.
2. Understand a wide variety of learning algorithms.
3. Understand how to evaluate models generated from data.
4. Apply the algorithms to a real problem, optimize the models learned and report on the expected accuracy that can be achieved by applying the models.
5. Understand the foundation of generative models.

Prerequisites:

- Design and Analysis of Algorithm
- Probability and Statistics

Detailed Content

<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><strong>Introduction to Machine Learning</strong>&lt;br&gt;Classical machine vs Adaptive Machine, Different Learning strategies, Supervised, Unsupervised, Semi Supervised, Reinforcement learning, Transfer learning (TL).&lt;br&gt;Basic of Training and Testing Phase Training and testing data, Over fitting and Under fitting</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td><strong>Feature Selection Techniques</strong>&lt;br&gt;Filter Methods- Information gain, Chi-Square test, Correlation and coefficient. Wrapper methods- Recursive feature elimination, Genetic Algorithm. Embedded method- Decision trees&lt;br&gt;Principal Component Analysis (PCA)</td>
<td>3</td>
</tr>
<tr>
<td>3</td>
<td><strong>Regression Analysis</strong>&lt;br&gt;Dependent vs Independent variable. Introduction, Confusion Matrix, Curse Of</td>
<td>7</td>
</tr>
</tbody>
</table>
### Textbook and Reference Books:


### PCC-AIML-504: Computer Networks

**Lecture per week (L): 3  Credits: 3**

### Course Outcomes (COs):

At the end of the course, the students will be able to:

1. **Design and investigate** why network needs flow control and error control techniques.
2. **Evaluate** the performance of the different routing protocol (RIP, OSPF) based on routing cost, convergence rate and complexity to find the shortest path.
3. **Analyze** the pieces of hardware (hub, bridge, switch, router) to make networks more efficient, faster, more secure, easier to use, able to transmit several simultaneous messages, and able to interconnect with other networks.
4. **Demonstrate** different LLC protocols, Internet Protocol, and usage of the IP address and subnet mask to setup a network.
5. **Understand** various techniques (open loop and close loop) used for congestion control and quality of service (traffic scheduling and shaping).
6. **Identify and remember** importance of existing protocols (DNS, DHCP, FTP, WWW, HTTP) running in application layer.

### Prerequisites:
Programming for Problem Solving
Data Structure and Algorithms
Computer Organization & Architecture

Detailed Content:

<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><strong>Introduction:</strong> Data communications, Direction of data flow - Simplex, Half-duplex, Full-duplex, Topology – Bus, Ring, Mesh. Star &amp; Hybrid, Types of Network - LAN, MAN 7 WAN, Protocols, and Reference models – OSI &amp; TCP/IP reference model &amp; comparative study.</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td><strong>Physical Layer:</strong> Transmission media - Guided &amp; Unguided, Switching – Circuit, Packet &amp; Message, Telephone Network, Network Devices: Repeaters, Hubs, Bridges, Switches, Router and Gateway. <strong>Data link Layer:</strong> Types of Errors, Error Detection – Parity, CRC &amp; Checksum, Error Correction – Hamming Code</td>
<td>6</td>
</tr>
<tr>
<td>3</td>
<td><strong>Data Link Layer and MAC Sublayer:</strong> Flow Control – Stop-n-Wait &amp; Sliding Window Protocol, ARQ Techniques – Stop-n-Wait, Go-Back- N &amp; Selective Repeat, Framing, Bit &amp; Byte Oriented Protocol, HDLC, Point to Point Protocol (PPP), Token Ring, FDDI and Ethernet Protocols, Reservation, Polling, Multiple access protocols - Pure ALOHA, Slotted ALOHA, CSMA, CSMA/CD, CSMA/CA.</td>
<td>8</td>
</tr>
<tr>
<td>4</td>
<td><strong>Network Layer:</strong> Internet Protocol (IP), IPv4 vs IPv6, ARP &amp; RARP, IP Addressing – Classful &amp; Classless, Subnetting, VLSM, CIDR. Routing - Techniques, Static, Dynamic &amp; Default Routing, Unicast Routing Protocols - RIP, OSPF, BGP.</td>
<td>5</td>
</tr>
<tr>
<td>5</td>
<td><strong>Transport Layer:</strong> Process to Process delivery; UDP; TCP; Congestion Control - Open Loop, Closed Loop, Quality of service, Techniques to improve QoS - Leaky bucket &amp; Token bucket algorithm.</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td><strong>Application Layer Protocols:</strong> DNS, SMTP, FTP &amp; DHCP. <strong>Modern Topic:</strong> Introduction to wireless LAN and Bluetooth, Mobile IP, Mobile TCP.</td>
<td>4</td>
</tr>
</tbody>
</table>

Text book and Reference books:

2) Data Communications and Networking (3rd Ed.) – B. A. Forouzan –TMH
3) Data and Computer Communications (5th Ed.) – W. Stallings –PHI/ Pearson Education
4) Computer Networking -A top down approach featuring the internet– Kurose and Rose –Pearson Education
5) Internetworking with TCP/IP, vol. 1, 2, 3(4th Ed.) – Comer –Pearson Education/PHI

PCC-AIML-505: Artificial Intelligence

Lecture per week (L): 3  Credits: 3

Course Outcomes (COs):

At the end of the course, the students will be able to:

1. Understand the various searching techniques, constraint satisfaction problem and example problems-game playing techniques.
2. Apply these techniques in applications which involve perception, reasoning and learning.
3. Explain the role of agents and how it is related to environment and the way of evaluating it and how agents can act by establishing goals.
4. Acquire the knowledge of real-world Knowledge representation.
5. Analyze and design a real-world problem for implementation and understand the dynamic behaviour of a system.
6. Use different machine learning techniques to design AI machine and enveloping applications for real world problems.

Prerequisites:
1. Data Structures
2. Probability

Detailed Content:

<table>
<thead>
<tr>
<th>MODULE NO</th>
<th>CONTENT</th>
<th>HOURS/MODULE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Introduction: Objective, scope and outcome of the course</td>
<td>1</td>
</tr>
<tr>
<td>3</td>
<td>Knowledge Representation, Problems in representing knowledge, knowledge representation using propositional and predicate logic, logical consequences, syntax and semantics of an expression, semantic Tableau. Forward and backward reasoning. Proof methods, substitution and unification, conversion to clausal form, normal forms, resolution, refutation, deduction, theorem proving, in referencing, monotonic and non-monotonic reasoning. Introduction to prolog.</td>
<td>8</td>
</tr>
<tr>
<td>4</td>
<td>Network-based representation and reasoning, Semantic networks, Conceptual Graphs, frames. Description logic (DL), concept language, reasoning using DL. Conceptual dependencies (CD), scripts, reasoning using CD. Introduction to natural language processing.</td>
<td>6</td>
</tr>
<tr>
<td>5</td>
<td>Adversarial search and Game theory, classification of games, game playing strategies, prisoner's Dilemma. Game playing techniques, minimax procedure, alpha-beta cut-offs. Complexity of alpha-beta search. Automated planning, classical planning problem, forward planning, partial order planning, planning with proposal logic, hierarchical task planning, multiagent planning</td>
<td>7</td>
</tr>
</tbody>
</table>

Text/Reference Books:

2. Introduction to AI & Expert System: Dan W.Patterson, PHI.
3. Artificial Intelligence by Luger (Pearson Education)
PCC-AIML-591: Operating System Lab
Contacts: 3P per week Credits: 1.5

Course Outcomes:

At the end of this course, students will demonstrate the ability to

- Demonstrate shell programming which include shell scripts and explaining shell syntax (variables, conditions, control structure, and functional commands).
- Execute programs like, creating a new process, creating orphan process and zombie process, synchronizing parent and child process.
- Analyze synchronization of co-operating processes with semaphore (semctl(), semget(), semop(), set semvalue, del semvalue, semaphore p and semaphore v).
- Adapt concept of signals with sending signals, signal interface, and signal handling.
- Apply POSIX threads using pthread_create, pthread_join and pthread_exit.
- Understand Inter-Process Communication (IPC) with use of pipes, message queue etc.

Detailed Content:

<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/ Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Shell programming: creating a script, making a script executable, shell syntax (variables, conditions, control structures, functions, commands).</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>Process: starting new process, replacing a process image, duplicating a process image, waiting for a process, zombie process.</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>Signal: signal handling, sending signals, signal interface, signal sets.</td>
<td>3</td>
</tr>
<tr>
<td>4</td>
<td>Semaphore: programming with semaphores (use functions semctl, semget, semop, set_semvalue, del_semvalue, semaphore_p, semaphore_v).</td>
<td>2</td>
</tr>
<tr>
<td>5</td>
<td>POSIX Threads: programming with pthread functions(viz. pthread_create, pthread_join, pthread_exit, pthread_attr_init, pthread_cancel)</td>
<td>3</td>
</tr>
<tr>
<td>6</td>
<td>Inter-process communication: pipes(use functions pipe, popen, pclose), named pipes(FIFOs, accessing FIFO)</td>
<td>3</td>
</tr>
</tbody>
</table>

Text and Reference Books:

1. UNIX Shell Programming by Yashavant Kanetkar, BPB Publication
3. UNIX and Shell Programming by Forouzan & Gilberg

PCC-AIML-592: Database Management System Lab
Contacts: 3P per week Credits: 1.5

Course Outcomes (COs):

At the end of the course the students are able to:

1. **Apply** the basic concepts of Database Systems and Applications.
2. **Define** the basics of SQL and construct queries using SQL in database creation and interaction.
3. **Design** a commercial relational database system (Oracle, MySQL) by writing SQL using the system.
4. **Analyze** and Select storage and recovery techniques of database system.
5. **Understand** various advanced queries execution such as relational constraints, joins, set operations, aggregate functions, trigger, views and embedded SQL.
6. **Construct** various software to design and build ER Diagrams, UML, Flow chart for related database systems.

**Prerequisites:**
- Data Base Management Systems

**Detailed Content:**

<table>
<thead>
<tr>
<th>Laboratory Experiments:</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Structured Query Language</strong></td>
</tr>
</tbody>
</table>

**1. Creating Database**
- Creating a Database
- Creating a Table
- Specifying Relational Data Types
- Specifying Constraints
- Creating Indexes

**2. Table and Record Handling**
- INSERT statement
- Using SELECT and INSERT together
- DELETE, UPDATE, TRUNCATE statements
- DROP, ALTER statements

**3. Retrieving Data from a Database**
- The SELECT statement
- Using the WHERE clause
- Using Logical Operators in the WHERE clause
  - Using IN, BETWEEN, LIKE, ORDER BY, GROUP BY and HAVING Clause
- Using Aggregate Functions
- Combining Tables Using JOINS
- Subqueries

**4. Database Management**
- Creating Views
- Creating Column Aliases
- Creating Database Users
- Using GRANT and REVOKE

**5. Cursors in Oracle PL / SQL**

**6. Writing Oracle PL / SQL Stored Procedures**

**PCC-AIML-593: Machine Learning Foundations Lab**
Labs per week: 3  
Credits: 1.5

**Course Outcomes (COs):**
At the end of the course, the students will be able to:

1. Recognize the characteristics of machine learning that make it useful to real-world problems.
2. Implement machine learning algorithms using toolboxes.
4. Analyze the performance of different classifiers.
5. Analyze the performance of different clustering algorithms.
6. Evaluate the machine learning models pre-processed through various feature engineering algorithms by python programming.

Prerequisites:
- IT Workshop (Python)

Detailed Content:

List of Assignments:
Lab 1: Write a program to demonstrate the working of Decision Tree algorithms. Use an appropriate dataset for building the decision trees and apply this knowledge to classify a new sample.
Lab 2: Write a program to demonstrate the working of Support Vector Machine (SVM) algorithms. Use an appropriate dataset and apply this knowledge to classify a new sample.
Lab 3: Write a program to implement k-Nearest Neighbor algorithm to classify a real life dataset. Print both correct and wrong predictions. Python ML library classes may be used for this problem.
Lab 4: Write a program to implement Random Forest algorithm to classify real life dataset.
Lab 5: Write a program to implement Naïve Bayes algorithm in real life dataset classification.
Lab 6: Write a program to implement K-means algorithm on real life dataset.
Lab 7: Write a program to implement linear and Logistic Regression in real life dataset.
Lab 8: Write a python program to implement Deep Learning in real life dataset.
Lab 9: Write a python program to implement PCA in real life dataset.
Lab10: Implementation of Artificial Neural Network.

Textbook and Reference books:


Semester - VI

PCC-AIML-601: Machine Learning Applications
Lectures per week: 2 Credits: 2

Course Outcomes (COs):

At the end of the course, the students will be able to:

1. Describe few Machine Learning systems like recommendation systems, social graph mining.
2. Implement ML algorithms to solve real world problems.
4. Design a machine learning system by incorporating various components of ML and evaluate the performance.
5. Develop machine learning model for bio informatics.
6. Understand the concept of Hidden Markov model.

Prerequisites:
- Machine Learning Foundations

Detailed Content:

<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Review of machine learning Concepts, Design of ML system – Model selection, bias, variance, learning curves, and error analysis</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>Recommendation Systems – Model for Recommendation Systems, Utility Matrix, Content- Based Recommendations, Discovering Features of Documents, Collaborative Filtering.</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>Mining Social network graphs – Clustering of Social Network Graphs, Partitioning of Graphs, and Finding Overlapping Communities</td>
<td>5</td>
</tr>
<tr>
<td>4.</td>
<td>Application of Machine learning algorithms in Bio- Informatics. Analyze the Biological data. Biological model development.</td>
<td>6</td>
</tr>
<tr>
<td>5.</td>
<td>Sparse models, State space models, Markov random Fields, Hidden Markov model, Review of Inference for graphical models, Latent Linear and Variable models for discrete data.</td>
<td>5</td>
</tr>
</tbody>
</table>

Text book and Reference books:


PCC-AIML-602: Deep Learning
Lectures per week: 3      Credits: 3

Course Outcomes (COs):

At the end of the course, the students will be able to:

1. Recognize the characteristics of deep learning models that are useful to solve real-world problems.
2. Understand different methodologies to create application using deep nets.
3. Identify and apply appropriate deep learning algorithms for analyzing the data for variety of problems.
4. Implement different deep learning algorithms.
5. Design the test procedures to assess the efficacy of the developed model.
6. Combine several models in to gain better results.

Prerequisites:
Detailed Content:

<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><strong>Introduction:</strong> Various paradigms of learning problems, Perspectives and Issues in deep learning framework, review of fundamental learning techniques</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td><strong>Feed forward neural network:</strong> Artificial Neural Network, activation function, multi-layer neural network, cardinality, operations, and properties of fuzzy relations</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td><strong>Training Neural Network:</strong> Risk minimization, loss function, back propagation, regularization, model selection, and optimization.</td>
<td>5</td>
</tr>
<tr>
<td>4</td>
<td><strong>Conditional Random Fields:</strong> Linear chain, partition function, Markov network, Belief propagation, Training CRFs, Hidden Markov Model, Entropy.</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td><strong>Transfer Learning</strong>&lt;br&gt;Delete learning Techniques, Variants of CNN: DenseNet, PixelNet.</td>
<td>3</td>
</tr>
<tr>
<td>6</td>
<td><strong>Deep Learning:</strong> Deep Feed Forward network, regularizations, training deep models, dropouts, Convolutional Neural Network, Deep Belief Network</td>
<td>7</td>
</tr>
<tr>
<td>7</td>
<td><strong>Auto Encoders</strong>&lt;br&gt;Under complete Auto encoder, Regularized Auto encoder, stochastic Encoders and Decoders, Contractive Encoders</td>
<td>4</td>
</tr>
</tbody>
</table>

Text book and Reference books:


HSMC-AIML-601: Human Values and Professional Ethics

Contacts: 3L per week  Credits: 3

Course Outcomes:

At the end of this course, students will demonstrate the ability to

- Understand Engineering and Technology as social and professional activities.
- Demonstrate the effects of technological growth, crisis of global resources and possible way out.
- Understand knowledge development for ethics in profession.
- Dissect development of professional and human values.
- Explain development of inner core and initiation of lifelong learning and survival process in professional arena.
- Demonstrate development of Moral character and thought of development of the country.

Detailed Content:
<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
</table>
| 1      | Science, Technology and Engineering as knowledge and as Social and Professional Activities  
**Effects of Technological Growth:**  
| 2      | **Ethics of Profession:**  
Engineering profession: Ethical issues in Engineering practice, Conflicts between business demands and professional ideals. Social and ethical responsibilities of Technologists. Codes of professional ethics. Whistle blowing and beyond, Case studies. | 8            |
| 3      | **Profession and Human Values:**  
Values Crisis in contemporary society. Nature of values: Value Spectrum of a good life. Psychological values: Integrated personality; mental health. Societal values: The modern search for a good society, justice, democracy, secularism, rule of law, values in Indian Constitution. Aesthetic values: Perception and enjoyment of beauty, simplicity, clarity. Moral and ethical values: Nature of moral judgments; canons of ethics; ethics of virtue; ethics of duty; ethics of responsibility. | 10           |

**Text book and Reference books:**

**PEC-AIML-601A: Soft Computing**  
**Lectures per week (L): 3  Credits: 3**

**Course Outcomes (COs):**
At the end of the course, the students will be able to:
1. Characterize supervised and unsupervised learning neural networks based on its architecture, training and testing mechanism.  
2. Apply the concept of fuzzification and defuzzification in fuzzy systems.  
3. Classify the architecture and working principles of specialized neural networks.  
4. Analyze the fundamental concepts of genetic algorithm and classify its types.  
5. Design, implement and evaluate a system / computer-based system, process, component or program to meet desired needs.  
6. Apply soft computing techniques to solve real time problems.

**Prerequisites:**
1. Mathematics  
2. Data Structure & Algorithms  
3. Programming and problem solving skills.
### Detailed Content:

<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Introduction: Introduction to soft computing; introduction to fuzzy sets and fuzzy logic systems; introduction to biological and artificial neural network; introduction to Genetic Algorithm.</td>
<td>2</td>
</tr>
<tr>
<td>5</td>
<td>Other Soft Computing techniques: Simulated Annealing, Tabu search, Ant colony optimization (ACO), Particle Swarm Optimization (PSO)</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td>Applications of Soft Computing: Image Fusion - Neural network classification - Traveling salesman problem using Genetic algorithm - Genetic algorithm based Internet searching technique.</td>
<td>4</td>
</tr>
</tbody>
</table>

### Text book and Reference books:

1. Fuzzy logic with engineering applications, Timothy J. Ross, John Wiley and Sons.
PEC-AIML-601B: Cloud Computing
Lectures per week (L): 3  Credits: 3

Course Outcomes (COs):

At the end of the course, the students will be able to:

1. Discuss the core concepts of cloud computing paradigm.
2. Analyze services, systems, platforms, frameworks to support cloud computing.
3. Illustrate the concepts of cloud storage system services.
4. Assess virtualization technology services in open source cloud computing environment.
5. Understand data center technology from industry centric perspective.
6. Identify cloud security issues to demonstrate real time applications.

Prerequisites:
1. Basic knowledge of Programming.
2. DBMS
4. Operating System.

Detailed Content:

<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Definition of Cloud Computing and Basics: Defining a Cloud, Cloud types- NIST Model, Cloud Cube model, Deployment models (Public, Private, Hybrid and Community Clouds), Service Platform as a Service, Software as a Service with examples of services/service providers, models – Infrastructure as a Service, Cloud Reference model, Characteristics of Cloud Computing – a shift in paradigm Benefits and advantages of Cloud Computing, A brief introduction on Composability, Infrastructure, Platforms, Virtual Appliances, Communication Protocols, Applications, Connecting to the Cloud by Clients, IaaS – Basic concept, Workload, partitioning of virtual private server instances, Pods, aggregations, silos PaaS – Basic concept, tools and development environment with examples SaaS - Basic concept and characteristics, Open SaaS and SOA, examples of SaaS platform Identity as a Service (IDaaS) Compliance as a Service (CaaS)</td>
<td>9</td>
</tr>
</tbody>
</table>
### Cloud Infrastructure:

- **Cloud Management:** An overview of the features of network management systems and a brief introduction of related products from large cloud vendors, Monitoring of an entire cloud computing deployment stack – an overview with mention of some products, Lifecycle management of cloud services (six stages of lifecycle).
- **Concepts of Cloud Security:** Cloud security concerns, Security boundary, Security service boundary, Overview of security mapping Security of data: Broketed cloud storage access, Storage location and tenancy, encryption, and auditing and compliance, Identity management (awareness of Identity protocol standards).

### Concepts of Services and Applications:

- **Service Oriented Architecture:** Basic concepts of message-based transactions, Protocol stack for an SOA architecture, Event-driven SOA, Enterprise Service Bus, Service catalogs, Applications in the Cloud: Concepts of cloud transactions, functionality mapping, Application attributes, Cloud service attributes, System abstraction and Cloud Bursting, Applications and Cloud APIs Cloud-based Storage: Cloud storage definition – Manned and Unmanned Webmail Services: Cloud mail services including Google Gmail, Mail2Web, Windows Live Hotmail, Yahoo mail, concepts of Syndication services.

### Data Center Technology:


---

**Text book and Reference books:**

6. Cloud Computing – Second Edition by Dr. Kumar Saurabh, Wiley India
PEC-AIML-601C: Pattern Recognition
Contact: 3L  Credits: 3

Course Outcomes:

At the end of this course, students will be able to:
1. Realize the clustering concepts and algorithms
2. Explain and compare a variety of pattern classification, structural pattern recognition, and pattern classifier combination techniques.
3. Summarize, analyze, and relate research in the pattern recognition area verbally and in writing.
4. Apply performance evaluation methods for pattern recognition, and critique comparisons of techniques made in the research literature.
5. Execute pattern recognition techniques to real-world problems such as document analysis and recognition.
6. Implement simple pattern classifiers, classifier combinations, and structural pattern recognizers.

Prerequisites:
1. Probability and linear algebra.
2. Data Mining.
3. Working knowledge of Matlab or Python

Detailed Content:

<table>
<thead>
<tr>
<th>Unit</th>
<th>Content</th>
<th>Hrs/Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Basics of pattern recognition</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>Bayesian decision theory Classifiers, Discriminant functions, Decision surfaces Normal density and discriminant functions Discrete features</td>
<td>8</td>
</tr>
<tr>
<td>3</td>
<td>Parameter estimation methods Maximum-Likelihood estimation Gaussian mixture models Expectation-maximization method Bayesian estimation</td>
<td>6</td>
</tr>
<tr>
<td>4</td>
<td>Hidden Markov models for sequential pattern classification Discrete hidden Markov models Continuous density hidden Markov models</td>
<td>8</td>
</tr>
<tr>
<td>5</td>
<td>Dimension reduction methods. Fisher discriminant analysis Principal component analysis. Parzen-window method K-Nearest Neighbour method</td>
<td>3</td>
</tr>
<tr>
<td>6</td>
<td>Non-parametric techniques for density estimation</td>
<td>2</td>
</tr>
<tr>
<td>7</td>
<td>Linear discriminant function based classifier Perceptron Support vector machines</td>
<td>5</td>
</tr>
<tr>
<td>8</td>
<td>Non-metric methods for pattern classification Non-numeric data or nominal data Decision trees</td>
<td>4</td>
</tr>
<tr>
<td>9</td>
<td>Unsupervised learning and clustering Criterion functions for clustering Algorithms for clustering: K-means, Hierarchical and other methods</td>
<td>2</td>
</tr>
</tbody>
</table>

Text book and Reference books:

PEC-AIML-601D: Graph Theory
Contact: 3L  Credits: 3

Course Outcomes (COs):

Upon completion of this course, the students should be able to:
1. Understand the various types of graph Algorithms and graph theory properties.
2. Analyze the NP – complete problems.
3. Distinguish the features of the various tree and matching algorithms
4. Appreciate the applications of digraphs and graph flow.
5. Understand the linear programming principles and its conversion.
6. Apply suitable graph model and algorithm for solving applications.

Prerequisites:
1. Mathematical proof technique (induction, proof by contradiction)
2. Linear algebra (determinants, eigen values).

Detailed Content:

<table>
<thead>
<tr>
<th>Unit</th>
<th>Content</th>
<th>Hrs/ Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Introduction - Graph Terminologies - Types of Graphs - Sub Graph- Multi Graph - Regular Graph - Isomorphism - Isomorphic Graphs - Sub-graph - Euler graph - Hamiltonian Graph - Related Theorems.</td>
<td>9</td>
</tr>
<tr>
<td>2</td>
<td>Trees -Properties- Distance and Centres - Types - Rooted Tree-- Tree Enumeration- Labeled Tree - Unlabeled Tree - Spanning Tree - Fundamental Circuits- Cut Sets - Properties - Fundamental Circuit and Cut-set- Connectivity- Separability -Related Theorems.</td>
<td>9</td>
</tr>
<tr>
<td>3</td>
<td>Network Flows - Planar Graph - Representation - Detection - Dual Graph - Geometric and Combinatorial Dual - Related Theorems - Digraph - Properties - Euler Digraph.</td>
<td>9</td>
</tr>
<tr>
<td>5</td>
<td>Graph Algorithms- Connectedness and Components- Spanning Tree- Fundamental Circuits- Cut Vertices- Directed Circuits- Shortest Path - Applications overview.</td>
<td>9</td>
</tr>
</tbody>
</table>

Text Books and Reference Books:

PEC AIML-602A: Big Data Analytics
Lecture per week: 3 Credits: 3

Course Outcomes (COs):

At the end of the course, the students will be able to:

1. **Understand** the concept of data management and evolution of Big Data.
2. **Understand** and implement various big data technology foundations.
3. **Apply** the fundamentals of Hadoop ecosystem and its components for data analysis.
4. **Analyze** the optimization techniques in data bases.
5. **Analyze** the storage techniques in data bases.
6. **Explore** the understanding of text, sentiment analytics.

Prerequisites:
- Database Management System
- C/C++ or Java in Linux
- Data Structures and Algorithms

Detailed Content:

<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
</table>
| 1      | **Introduction to Big Data and Hadoop**  
Types of Digital Data, Introduction to Big Data, Big Data Analytics, History of Hadoop, Apache Hadoop, Analysing Data with Unix tools, Analysing Data with Hadoop, Hadoop Streaming, Hadoop Echo System, IBM Big Data Strategy, Introduction to Infosphere Big Insights and Big Sheets | 5 |
| 2      | **HDFS(Hadoop Distributed File System)**  
The Design of HDFS, HDFS Concepts, Command Line Interface, Hadoop file system interfaces, Data flow, Data Ingest with Flume and Scoop and Hadoop archives, Hadoop I/O: Compression, Serialization, Avro and File-Based Data structures. | 7 |
| 3      | **Map Reduce**  
Anatomy of a Map Reduce Job Run, Failures, Job Scheduling, Shuffle and Sort, Task Execution, Map Reduce Types and Formats, Map Reduce Features. | 7 |
| 4      | **Hadoop Eco System**  
Pig :  
Introduction to PIG, Execution Modes of Pig, Comparison of Pig with Databases, Grunt, Pig Latin, User Defined Functions, Data Processing operators.  
Hive :  
Hive Shell, Hive Services, Hive Metastore, Comparison with Traditional Databases, HiveQL, Tables, Querying Data and User Defined Functions.  
Hbase : HBasics, Concepts, Clients, Example, Hbase Versus RDBMS.  
Big SQL : Introduction | 7 |
| 5      | **Data Analytics with R**  
Machine Learning: Introduction, Supervised Learning, Unsupervised Learning, Collaborative Filtering. Big Data Analytics with BigR | 6 |

Text book and Reference books:
PEC-AIML-602B: Data Mining

Lecture per week (L): 3    Credits: 3

Course Outcomes (COs):

At the end of the course, the students will be able to:

1. Understand what is Data Mining, what kinds of data can be mined, what kinds of patterns can be mined, and what kinds of applications are targeted.
2. Explain major issues in data mining.
3. Apply machine learning, pattern recognition, statistics, visualization, algorithm, database technology and high-performance computing in data mining applications.
4. Identify what kinds of technologies are used for different applications.
5. Manipulate data preprocessing, data warehouse and OLAP technology, data cube technology; mining frequent patterns and association, classification, clustering, and outlier detection.
6. Discover interesting patterns from large amounts of data to analyze for predictions and classification.

Prerequisites:
- Basic understanding of DBMS
- Engineering Mathematics
- Data Structure and Algorithm

Detailed Content:

<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Introduction: Evolution and Importance of Data Mining-Types of Data and Patterns Mined-Technologies-Applications-Major Issues in Data Mining. Knowing about Data-Data Pre-processing: Cleaning–Integration–Reduction–PCA, Data Transformation and Discretization. Mining Frequent Patterns: Basic Concept – Frequent Item Set Mining Methods – Mining Association Rules – Association to Correlation Analysis.</td>
<td>8</td>
</tr>
<tr>
<td>2</td>
<td>Classification and Prediction: Issues – Decision Tree Induction – Bayesian Classification – Rule Based Classification – k-Nearest-Neighbor Classification – Linear SVM – Regression – Linear, Logistic – Accuracy and Error measures – Introduction to Ensemble methods</td>
<td>6</td>
</tr>
<tr>
<td>3</td>
<td>Clustering: Overview of Clustering – Types of Data in Cluster Analysis – Major Clustering Methods-Partitioning Methods- k-Means, k-Medoids, Hierarchical Methods-Agglomerative and Divisive hierarchical clustering. Density-Based Methods-DBSCAN, Graph-based clustering (CHAMELEON), Evaluation in Clustering</td>
<td>6</td>
</tr>
</tbody>
</table>
### Text book and Reference books:

1. Jiawei Han, Micheline Kamber and Jian Pei, “Data mining concepts and Techniques”, Third Edition, Elsevier Publisher, 2006.
5. Itay Lieder, Yehezkel Resheff, Tom Hope, Learning TensorFlow, O’Reilly Media, 2017

### PEC-AIML-602C: Distributed System

**Lecture per week:** 3  
**Credits:** 3

#### Course Outcomes (COs):

At the end of the course, the students will be able to:

1. **Understand the** basic knowledge of Distributed Computing.
2. **Understand** the Distributed Models.
3. **Compare** interposes communication and remote communication.
4. **Remember** the concept of service oriented architecture.
5. **Apply different** emerging techniques in distributed computing.
6. **Design** Distributed Shared Memory and File System.

#### Prerequisites:

- Computer Networks
- Operating System

#### Detailed Content:

<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Basic Network Communication: LAN and WAN Technologies, Classification of Networks, Protocols for Network Systems, ATM, Protocols for Distributed Systems</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>Inter process and Remote Communication:</td>
<td>5</td>
</tr>
</tbody>
</table>
Message Passing, IPC in Mach, CBCAST protocol in ISIS, RPC Introduction and Basics, RPC Implementation and Communication, Sun RPC, RMI Implementation

Distributed System Synchronization:
Introduction, Clock Synchronization, Logical and Global state, Mutual Exclusion, Election Algorithms, Deadlocks in Distributed Systems, Deadlocks in Message Communication

5

Distributed System Management:

5

Distributed Shared Memory:

5

Distributed File System:

4

Text book and Reference books:
1. Distributed Computing, Sunita Mahajan and Seema Shah, Oxford University
2. Distributed Operating Systems by P. K. Sinha, PHI
4. Distributed Systems: Principles and Paradigms, Taunenbaum

PEC-AIML-602D: Digital Signal Processing
Lecture per week (L): 3 Credits: 3

Course Outcomes (COs):
At the end of the course, the students will be able to:
1. Design linear discrete-time systems and filters and analyse their behaviour.
2. Represent continuous-time signals and linear systems in discrete time, so that such signals can be recovered in continuous time when necessary.
3. Compute approximations to Fourier transforms of continuous-time signals with finite discrete time methods.
4. Understand the analytical tools such as Fourier transforms, Discrete Fourier transforms, Fast Fourier Transforms and Z-Transforms required for digital signal processing.
5. Design and realize various digital filters for digital signal processing.
6. Understand the applications of DSP in speech processing and spectrum analysis.

Prerequisites:
- Digital Electronics

Detailed Content:
<table>
<thead>
<tr>
<th>Sl No</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Introduction: Objective, scope and outcome of the course.</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>Discrete-time signals:</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>Concept of discrete-time signal, basic idea of sampling and reconstruction of signal, sampling theorem, sequences – periodic, energy, power, unit-sample, unit-step, unit-ramp, real &amp; complex exponentials, arithmetic operations on sequences.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>LTI Systems:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Definition, representation, impulse response, derivation for the output sequence, concept of convolution, graphical, analytical and overlap-add methods to compute convolution supported with examples and exercises, properties of convolution, interconnections of LTI systems with physical interpretations, stability and causality conditions, recursive and non-recursive systems.</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Z-Transform:</td>
<td>11</td>
</tr>
<tr>
<td></td>
<td>Definition, mapping between s-plane and z-plane, unit circle, convergence and ROC, properties of Z-transform, Z-transform on sequences with examples and exercises, characteristic families of signals along with ROCs, convolution, correlation and multiplication using Z-transform, initial value theorem, Perseval’s relation, inverse Z-transform by contour integration, power series &amp; partial-fraction expansions with examples and exercises.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Discrete Fourier Transform:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Concept and relations for DFT/IDFT, Twiddle factors and their properties, computational burden on direct DFT, DFT/IDFT as linear transformations, DFT/IDFT matrices, computation of DFT/IDFT by matrix method, multiplication of DFTs, circular convolution, computation of circular convolution by graphical, DFT/IDFT and matrix methods, linear filtering using DFT, aliasing error, filtering of long data sequences – Overlap-Save and Overlap-Add methods with examples and exercises.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Fast Fourier Transform:</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Filter Design:</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Basic concepts of IIR and FIR filters, difference equations, design of Butterworth IIR analog filter using impulse invariant and bilinear transforms, design of linear phase FIR filters, no. of taps, rectangular, Hamming and Blackman windows.</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Digital Signal Processor:</td>
<td>5</td>
</tr>
<tr>
<td></td>
<td>Elementary idea about the architecture and important instruction sets of TMS320C 5416/6713 processor, writing of small programs in Assembly Language.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>FPGA:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Architecture, different sub-systems, design flow for DSP system design, mapping of DSP algorithms onto FPGA.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TOTAL</td>
<td>30</td>
</tr>
</tbody>
</table>

**Text and reference books:**

OEC-AIML-601A: Advanced Computer Architecture

Contacts: 3L per week
Credits: 3

Course Outcomes:
At the end of this course, students will demonstrate the ability to
- Explain various parameters to measure the performance of a processor.
- Demonstrate the parallel computing concepts and compare parallel computing with sequential computing.
- Explain the pipelining technique and its related issues.
- Demonstrate the vector processing, array processors and multiprocessors.
- Outline and design various types of interconnection networks for parallel computers.
- Dissect different techniques required to improve the performances of cache memory and main memory.

Prerequisites:
Computer Organization & Architecture

Detailed Content:

<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Introduction: Review of basic computer architecture (Revisited), Quantitative techniques in computer design, measuring and reporting performance.</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>Pipelining: Basic concepts, instruction and arithmetic pipeline, data hazards, control hazards and structural hazards, techniques for handling hazards. Exception handling. Pipeline optimization techniques; Compiler techniques for improving performance.</td>
<td>9</td>
</tr>
<tr>
<td>3</td>
<td>Instruction-level parallelism: basic concepts, techniques for increasing ILP, superscalar, super-pipelined and VLIW processor architectures. Array and vector processors.</td>
<td>6</td>
</tr>
<tr>
<td>4</td>
<td>Multiprocessor architecture: taxonomy of parallel architectures; Centralized shared-memory architecture: synchronization, memory consistency, interconnection networks. Distributed shared-memory architecture. Cluster computers.</td>
<td>8</td>
</tr>
<tr>
<td>5</td>
<td>Non von Neumann architectures: data flow computers, reduction computer architectures, systolic architectures.</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td>Hierarchical memory technology: Inclusion, Coherence and locality properties; Cache memory organizations, Techniques for reducing cache misses; Virtual memory organization, mapping and management techniques, memory replacement policies.</td>
<td>8</td>
</tr>
</tbody>
</table>

Text book and Reference books:
1. Computer Architecture and Parallel Processing by Hwang and Briggs (Mc-Graw Hill)
3. Computer Architecture: A Qualitative Approach by Hennesey & Patterson (Morgan Kaufman)

OEC-AIML-601B: Human Computer Interaction
Lecture per week (L): 3   Credits: 3

Course Outcomes (COs):
At the end of the course, the students will be able to:
1. **Explain** the capabilities of both humans and computers from the viewpoint of human information processing.

2. **Describe** typical human–computer interaction (HCI) models, styles, and various historic HCI paradigms.

3. **Apply** an interactive design process and universal design principles for designing HCI systems.

4. **Analyze** and identify user models, user support, socio-organizational issues, and stakeholder requirements of HCI systems.

5. **Apply** design and development principles in the construction of HCI systems

6. **Use** current techniques, skills, and tools necessary for computing practice.

**Prerequisites:**

- Basics of programming
- Data Structure and Algorithms

**Detailed Content:**

<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Introduction to Human-Computer Interaction, Task-centered system design: task-centered process, development of task examples, evaluation of designs through a task-centered walk-through</td>
<td>6</td>
</tr>
<tr>
<td>2</td>
<td>User-centered design and prototyping: assumptions, participatory design, methods for involving the user, prototyping, low fidelity prototypes, medium fidelity</td>
<td>6</td>
</tr>
<tr>
<td>3</td>
<td>Methods for evaluation of interfaces with users: goals of evaluation, approaches, ethics, introspection, extracting the conceptual model, direct observation, constructive interaction, interviews and questionnaires, continuous evaluation via user feedback and field studies, choosing an evaluation method</td>
<td>8</td>
</tr>
<tr>
<td>4</td>
<td>Psychology of everyday things: psychopathology of everyday things, examples, concepts for designing everyday things, Beyond screen design: characteristics of good representations, information visualization, Tufte’s guidelines, visual variables</td>
<td>5</td>
</tr>
<tr>
<td>5</td>
<td>Graphical screen design: graphical design concepts, Design principles and usability heuristics: design principles, principles to support usability, HCI design standards: process-oriented standards, product-oriented standards, Past and future of HCI: the past, present and future, perceptual interfaces, context-awareness and perception.</td>
<td>5</td>
</tr>
</tbody>
</table>

**Text book and Reference books:**


**OEC-AIML-601C: Artificial Neural Network**

**Lecture per week (L): 3 \quad Credits: 3**

**Course Outcomes (COs):**

At the end of the course, the students will be able to:

1. **Analyze** synaptic connectivity as the basis of neural computation and learning
2. Evaluate the ideological basics of artificial neural networks
3. Understand the origins of artificial neural networks
4. Know some application of artificial neural networks
5. Identify the different structures of artificial neural networks.
6. Learn perceptron and dynamical theories of recurrent networks including amplifiers, attractors, and hybrid computation.

Prerequisites:
- Higher Engineering Mathematics e.g. linear algebra, multivariate calculus and Probability theory,
- Data Structure and Algorithms
- Fundamental knowledge of signals and systems along with types, Mathematical representation of signals and system modelling in time as well as frequency domain.

Detailed Content:

<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Introduction to ANN Features, structure and working of Biological Neural Network, Trends in Computing Comparison of BNN and ANN</td>
</tr>
<tr>
<td>3</td>
<td>Backpropagation networks : (BPN) Architecture of feed forward network, single layer ANN, multilayer perceptron, back propagation learning, input - hidden and output layer computation, backpropagation algorithm, applications, selection of tuning parameters in BPN, Numbers of hidden nodes, learning.</td>
</tr>
<tr>
<td>4</td>
<td>Activation &amp; Synaptic Dynamics: Introduction, Activation Dynamics models, synaptic Dynamics models, stability and convergence, recall in neural networks.</td>
</tr>
<tr>
<td>5</td>
<td>Basic functional units of ANN for pattern recognition tasks: Basic feed forward, Basic feedback and basic competitive learning neural network.</td>
</tr>
</tbody>
</table>

Text book and Reference books:

1. B. Yegnanarayana - Artificial neural network PHI Publication.
2. S. Raj sekaran, Vijayalakshmi Pari - Neural networks, Fuzzy logic and Genetic Algorithms

OEC-AIML-601D: Cryptography and Network Security
Lecture per week (L): 3 Credits: 3
Course Outcomes (COs):

At the end of the course, the students will be able to:

1. **Design** network application security schemes, such as PGP, S/MIME, SSL, HTTPS etc.
2. **Evaluate** concepts of Firewall (including types of Firewall), DMZ Network and comparing between different Firewall Configurations.
3. **Analyze** Biometric Authentication and differentiate between different types of Authentication tokens.
4. **Implement and apply** numerical module based on DES and RSA illustrating the concept of SSL, PGP, Authentication token, Digital Signature, Message Digest and Hash function.
5. **Understand** and Classify different kinds of Substitution techniques and Transposition techniques and discuss the concepts of Symmetric key cryptography and Asymmetric key cryptography.
6. **Define** the concepts of Network security and identifying different types of attack on Network security.

Prerequisites:

- Computer Networks
- Discrete Mathematics

Detailed Content:

<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td><strong>Symmetric Key Algorithms</strong>: Algorithm types &amp; Modes, Overview of Symmetric Key Cryptography, Diffie-Hellman key exchange algorithm, Digital Envelope, DES(Data Encryption Standard) algorithm &amp; its variant, IDEA(International Data Encryption Algorithm) algorithm, RC5(Rivest Cipher 5) algorithm.</td>
<td>8</td>
</tr>
<tr>
<td>3</td>
<td><strong>Asymmetric Key Algorithms</strong>: Digital Signature and User Authentication, Overview of Asymmetric key Cryptography, RSA algorithm, Digital Signature, Basic concepts of Message Digest and Hash Function (Algorithms on Message Digest and Hash function not required), HMAC algorithm. Authentication Basics, Password, Authentication Token, Certificate based Authentication and Biometric Authentication.</td>
<td>9</td>
</tr>
<tr>
<td>4</td>
<td><strong>Electronic mail security, SSL and Firewall</strong>: Basics of mail security, PEM, PGP, S/MIME, Secure Socket Layer (SSL) protocol. Introduction to Firewall, Types of firewall, Firewall Configurations and DMZ Network.</td>
<td>6</td>
</tr>
</tbody>
</table>

Text book and Reference books:


PCC-AIML-691: Machine Learning Applications Lab
Labs per week: 3 Credits: 1.5

Course Outcomes (COs):

At the end of the course, the students will be able to:
1. **Develop** Machine learning models.
2. **Apply** the knowledge of machine learning in various practical fields.
3. **Design** machine learning algorithms to solve real life problems.
4. **Understand** different machine learning techniques.
5. **Analyze** and compare the performance of different Machine learning algorithms.

Prerequisites:
- Machine Learning Foundations
- Machine Learning Applications
- IT Workshop (Python)

Tentative List of Experiments:

1. Implementation of Recommendation Systems.
2. Predictive Analysis on real life data.

Text book and Reference books:


PCC-AIML-692: Artificial Intelligence Lab
Labs per week (L): 3 Credits: 1.5

Course Outcomes (COs):

At the end of the course, the students will be able to:
1. Demonstrate the applications of AI and agent-based approach to AI.
2. Obtain first-order predicate calculus, logical reasoning and problem solving using Python language.
3. Study and discuss various techniques and algorithms of AI used in general problem solving, optimization problems, constraint satisfaction problems, and game programming.
4. Familiarize students with various sub-areas of AI, such as expert systems, natural language processing and machine learning.
5. Study and discuss various techniques and algorithms of AI used in Genetic Algorithm.
6. Dissect various techniques and algorithms of AI used in Expert System.

**Prerequisites:**
- Data Structures
- Probability
- Python Programming Language

**Detailed Content:**

<table>
<thead>
<tr>
<th>Sl No</th>
<th>Content</th>
</tr>
</thead>
</table>
| 1 | Anaconda:  
Learn how to use Anaconda to manage packages and environments for use with Python. |
| 2 | Jupyter Notebooks:  
Learn how to use Jupyter Notebooks to create documents combining code, text, images, and more. |
| 3 | Numpy Basics:  
• Learn the value of NumPy and how to use it to manipulate data for AI problems.  
• Mini-Project: Use NumPy to mean normalize and array and separate it into several smaller arrays. |
| 4 | Pandas Basics:  
• Learn to use Pandas to load and process data for machine learning problems.  
• Mini-Project: Use Pandas to plot and get statistics from stock data |
| 5 | Matplotlib Basics:  
Learn how to use Matplotlib to choose appropriate plots for one and two variables based on the types of data you have. |

**Text/Reference Books:**

1. Machine Learning for Absolute Beginners: A Plain English Introduction Author: Oliver Theobald Publisher — Scatterplot Press
3. Machine Learning (in Python and R) For Dummies Author: John Paul Mueller and Luca Massaron
4. Machine Learning in Action Author: Peter Harrington Publisher — Manning Publications

**PCC-AIML-693: Computer Networks Lab**

Labs per week (P): 3  Credits: 1.5

**Course Outcomes (COs):**

At the end of the course, the students will be able to:
1. **Design** an application to execute command remotely using socket programming.
2. **Evaluate** file transfer application using socket programming.
3. **Analyze** the hardware (hub, bridge, switch, router) to make networks more efficient, faster, more secure, easier to use, able to transmit several simultaneous messages, and able to interconnect with other networks.
4. **Implement** error-control mechanism for data transmission.
5. **Understand** the concepts of NIC installation and configuration.
6. **Remember** to gather network information using socket programming.

**Prerequisites:**
- Programming for Problem Solving
- Data Structure and Algorithms
- Object Oriented Programming & Java
- Computer Networks

**Detailed Content:**

1. NIC Installation & Configuration
2. TCP/UDP Socket Programming – Introduction
3. Sockets – Operation, Socket types, Domains, Closing Sockets
4. Client/Server Models - Usage
5. Connection Based Services - Client and Server actions
6. Connectionless Services - Client and Server actions
7. Access Network Database - Host Information, Network Information, Protocol Information
8. Implement Multicasting / Broadcasting socket I/O.
9. Implement ARQ techniques.

**Text book and Reference books:**

2) Data Communications and Networking (3rd Ed.) – B. A. Forouzan –TMH
3) Data and Computer Communications (5th Ed.) – W. Stallings –PHI/ Pearson Education
4) Computer Networking - A top down approach featuring the internet – Kurose and Rose – Pearson Education
5) Internetworking with TCP/IP, vol. 1, 2, 3(4th Ed.) – Comer –Pearson Education/PHI

=================================================================================

**Semester - VII**

**PEC-AIML-701A: Social Network Analysis**

Lecture per week (L): 3 
Credits: 3

**Course Outcomes (COs):**

At the end of the course, the students will be able to:
1. Understand the concept of semantic web and related applications.
2. Learn knowledge representation using ontology.
4. Learn visualization of social networks.
5. Develop semantic web related applications.
Prerequisites:
- Basic Mathematics
- Probability and Statistics

Detailed Content:

<table>
<thead>
<tr>
<th>Sl No</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Introduction: Objective, scope and outcome of the course.</td>
<td>1</td>
</tr>
<tr>
<td>2</td>
<td>Introduction to Semantic Web: Limitations of current Web - Development of Semantic Web - Emergence of the Social Web - Social Network analysis: Development of Social Network Analysis - Key concepts and measures in network analysis - Electronic sources for network analysis: Electronic discussion networks, Blogs and online communities - Web-based networks - Applications of Social Network Analysis.</td>
<td>5</td>
</tr>
<tr>
<td>5</td>
<td>Understanding and predicting human behaviour for social communities - User data management - Inference and Distribution - Enabling new human experiences - Reality mining - Context - Awareness - Privacy in online social networks - Trust in online environment - Trust models based on subjective logic - Trust network analysis - Trust transitivity analysis - Combining trust and reputation - Trust derivation based on trust comparisons - Attack spectrum and countermeasures.</td>
<td>6</td>
</tr>
<tr>
<td>6</td>
<td>Graph theory - Centrality - Clustering - Node-Edge Diagrams - Matrix representation - Visualizing online social networks, Visualizing social networks with matrix-based representations - Matrix and Node-Link Diagrams - Hybrid representations - Applications - Cover networks - Community welfare - Collaboration networks - Co-Citation networks.</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>Total</td>
<td>33</td>
</tr>
</tbody>
</table>

Text/ References Books:

PEC-AIML-701B: Computer Vision
Lecture per week (L): 3 Credits: 3

Course Outcomes (COs):

Upon completion of the course, students will be able to
1. Understand and master basic knowledge, theories and methods in image processing and computer vision.
2. Identify, formulate and solve problems in image processing and computer vision.
3. Analyze, evaluate and examine existing practical computer vision systems.
4. Critically review and assess scientific literature in the field and apply theoretical knowledge to identify the novelty and practicality of proposed methods.
5. Design and develop practical and innovative image processing and computer vision applications or systems.
6. Conduct themselves professionally and responsibly in the areas of computer vision image processing and deep learning.

Prerequisites:
1. Mathematics (Linear algebra, vector calculus, and probability)
2. Data structures
3. Programming

Detailed Content:

<table>
<thead>
<tr>
<th>Unit</th>
<th>Contents</th>
<th>Hrs/Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Digital Image Formation and low-level processing: Overview and State-of-the-art, Fundamentals of Image Formation, Transformation: Orthogonal, Euclidean, Affine, Projective, etc; Fourier Transform, Convolution and Filtering, Image Enhancement, Restoration, Histogram Processing, introduction to computer vision.</td>
<td>8</td>
</tr>
<tr>
<td>2</td>
<td>Feature Extraction: Shape, histogram, color, spectral, texture, Feature analysis, feature vectors, distance /similarity measures, data preprocessing, Edges - Canny, LOG, DOG; Scale-Space Analysis- Image Pyramids and Gaussian derivative filters, Gabor Filters and DWT; Line detectors (Hough Transform), Orientation Histogram, SIFT, SURF, GLOH, Corners - Harris and Hessian Affine.</td>
<td>8</td>
</tr>
<tr>
<td>3</td>
<td>Depth estimation and Multi-camera views: Perspective, Homography, Rectification, DLT, RANSAC, 3-D reconstruction framework; Binocular Stereopsis: Camera and Epipolar Geometry; Auto-calibration. Image Segmentation: Region Growing, Edge Based approaches to segmentation, Graph-Cut, Mean-Shift, MRFs, Texture Segmentation; Object detection.</td>
<td>8</td>
</tr>
<tr>
<td>4</td>
<td>Motion Analysis: Optical Flow, KLT, Spatio-Temporal Analysis, Background Subtraction and Modeling, Dynamic Stereo; Motion parameter estimation.</td>
<td>6</td>
</tr>
<tr>
<td>5</td>
<td>Shape from X: Light at Surfaces; Use of Surface Smoothness Constraint; Shape from Texture, color, motion and edges Albedo estimation; Photometric Stereo; Phong Model; Reflectance Map.</td>
<td>6</td>
</tr>
</tbody>
</table>

Text and Reference Books:
PEC-AIML-701C: Software Engineering

Lecture per week (L): 3  Credits: 3

Course Outcomes (COs):

At the end of the course, the students will be able to:

1. Describe the importance of software engineering lifecycle models in the development of software and apply the knowledge to the solution of complex engineering problems.
2. Analyze the requirements and develop SRS documents following the principles in modeling software based on real life applications.
3. Design and develop software which adheres to the standard software design guideline & benchmarks.
4. Create and apply appropriate techniques for software testing.
5. Demonstrate knowledge and understanding of engineering and management principles for software projects management.
6. Understand software matrices like size, effort and cost estimation, and software quality metrics.

Prerequisites:

- Basic knowledge of programming

Detailed Content:

<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Structured Analysis, Context diagram and DFD, Physical and Logical DFDs, Data Modelling, ER diagrams, Software Requirements Specification.</td>
<td>5</td>
</tr>
<tr>
<td>3</td>
<td>Design Aspects: Top-Down and Bottom-Up design; Decision tree, decision table and structured English, Structure chart, Transform analysis Functional vs. Object-Oriented approach.</td>
<td>6</td>
</tr>
<tr>
<td>4</td>
<td>Unified Modeling Language Class diagram, interaction diagram: collaboration diagram, sequence diagram, state chart diagram, activity diagram, implementation diagram.</td>
<td>4</td>
</tr>
<tr>
<td>5</td>
<td>Coding &amp; Documentation - Structured Programming, Modular Programming, Module Relationship- Coupling, Cohesion, OO Programming, Information Hiding, Reuse, System Documentation.</td>
<td>5</td>
</tr>
<tr>
<td>6</td>
<td>Testing - Levels of Testing, Integration Testing, System Testing, Software Quality.</td>
<td>9</td>
</tr>
</tbody>
</table>
Text and Reference books:

1. Software Engineering: A practitioner’s approach - Pressman (TMH)
2. Software Engineering- Pankaj Jalote (Wiley-India)
3. Software Engineering- Rajib Mall (PHI)
4. Software Engineering -Agarwal and Agarwal (PHI)

PEC-AIML-701D: Data Warehousing
Lecture per week (L): 3 Credits: 3

Course Outcomes (COs):

At the end of the course, the students will be able to:

1. Define the knowledge of mathematics and science on data warehouse, building blocks, Data Mart and recall in independent and life-long learning of data warehouse.
2. Classify Data warehouse Architecture in the areas of Data acquisition, Data storage and Information delivery and illustrate the engineering principles.
3. Make use of the architecture and infrastructure of Database Software and model appropriate tools for database software by applying the knowledge of software development by individual or team.
4. Analyze Metadata types by functional areas and assume effective reports on Business metadata by understanding of the engineering principles of metadata.
5. Justify effective reports on Data acquisition, Data storage, and Information delivery and evaluate the ability for life-long learning on data storage.
6. Discuss Knowledge Discovery Process, OLAP, Different techniques by building the knowledge of mathematics and engineering fundamentals on OLAP and develop applications in societal, health, safety, legal and cultural issues.

Prerequisites:
1. Knowledge of Programming skill.
2. Basic Statistics and mathematics.

Detailed Content:

<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Data Warehouse: Basic Concepts, Differences between Operational Database Systems and Data Warehouses- A Multitiered Architecture - Data Warehouse Models : Extraction, Transformation and Loading - Metadata Repository -Data Cube and OLAP -Data Warehouse Design and Usage – Data warehouse implementation.</td>
<td>7</td>
</tr>
<tr>
<td>2</td>
<td>Introduction To Data Mining: Introduction - The evolution of database system technology - Steps in knowledge discovery from database process - Architecture of a data mining systems - Data mining on different kinds of data - Different kinds of pattern - Technologies used - Applications - Major issues in data mining - Classification of data mining systems – Data mining task primitives - Integration of a data mining system with a database or data warehouse system.</td>
<td>8</td>
</tr>
<tr>
<td>3</td>
<td>Data Preprocessing: Data Objects and attribute types - Basic statistical description</td>
<td>8</td>
</tr>
</tbody>
</table>
of data - Data visualization – Measuring data similarity and dissimilarity - Data cleaning - Integration - Data reduction – Data transformation and data discretization.

4 Association Rule Mining: Basic concepts - Frequent itemset mining methods - Apriori algorithm, APattern growth approach for mining frequent itemsets, Mining frequent itemsets using vertical data format, Mining closed and max patterns - Pattern mining in multilevel and multidimensional space – Constraint based Frequent pattern mining - Mining High-Dimensional Data and Colossal Patterns 8

5 Classification And Clustering: Classification : Basic concepts - Decision tree induction - Bayes classification methods-Rule Based Classification- Model Evaluation and Selection - Techniques to Improve Classification Accuracy - Bayesian Belief Networks - Classification by Back propagation - Cluster Analysis – Partitioning methods- Hierarchical methods. 9

Text and Reference books:

1. Jiawei Han, Micheline Kamber and Jian Pai, Data Mining: Concepts and Techniques, Morgan Kauffman, 2013
2. Alex Berson and Stephen J Smith, Data Warehousing, Data Mining, and OLAP, Mcgraw- Hill,2008
4. Margaret H. Dunham, Data Mining: Introductory and Advanced Topics, Pearson Education 2008

PEC-AIML-702A: Ecommerce and ERP

Lecture per week (L): 3 Credits: 3

Course Outcomes (COs):

At the end of the course, the students will be able to:
1. Understand fundamental concepts and technologies related to ERP.
2. Explain the different phases of ERP implementation life cycle.
3. Examine the modules, benefits and various tools of ERP.
4. Analyze the impact of e-commerce on business model and strategies.
5. Assess the electronic payment systems and software.
6. Identify and solve the security issues related to communication.

Prerequisites:
• Basic Mathematics

Detailed Content:

<table>
<thead>
<tr>
<th>Sl No</th>
<th>Content</th>
<th>Hours/ Module</th>
</tr>
</thead>
</table>
### Telecommunication.

<table>
<thead>
<tr>
<th>Sl No</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>E-Commerce Payment and Gateways&lt;br&gt;Electronic Payments Overview of Electronics payments, Digital Token based Electronics payment System, Smart Cards, Credit Card I Debit Card based EPS, Emerging financial Instruments, Home Banking, Online Banking.</td>
<td>6</td>
</tr>
<tr>
<td>4</td>
<td>E-Commerce and EDA&lt;br&gt;Net Commerce EDA, EDI Application in Business, Legal requirement in E-Commerce, Introduction to supply Chain Management, CRM, issues in Customer Relationship Management.</td>
<td>6</td>
</tr>
<tr>
<td>5</td>
<td>Internet and E-Commerce&lt;br&gt;Internet and Electronic commerce, internet, extranet and enterprise solutions, information system for business operations, information system for managerial decision support, information system for strategic advantage.</td>
<td>6</td>
</tr>
<tr>
<td></td>
<td>Total</td>
<td>30</td>
</tr>
</tbody>
</table>

### Books Recommended:

### PEC-AIML-702B: Information Theory and Coding

Lecture per week (L): 3  Credits: 3

### Course Outcomes (COs):

At the end of the course, the students will be able to:
1. Apply various source coding techniques.
2. Design the channel performance using information theory.
3. Comprehend various error control code properties
4. Apply linear block codes for error detection and correction
5. Apply convolution codes for performance analysis & cyclic codes for error detection and correction.
6. Design BCH & RS codes for channel performance improvement against burst errors.

### Prerequisites:
1. Basic Mathematics
2. Probability and Statistics

### Detailed Content:

<table>
<thead>
<tr>
<th>Sl No</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Introduction: Objective, scope and outcome of the course.</td>
<td>1</td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
</tbody>
</table>
| 2 | Source Coding  
Uncertainty and information, average mutual information and entropy, information measures for continuous random variables, source coding theorem, Huffman codes. | 5 |
| 3 | Channel Capacity And Coding  
Channel models, channel capacity, channel coding, information capacity theorem, The Shannon limit. | 5 |
| 4 | Linear And Block Codes For Error Correction  
Matrix description of linear block codes, equivalent codes, parity check matrix, decoding of a linear block code, perfect codes, Hamming codes. | 5 |
| 5 | Cyclic Codes  
Polynomials, division algorithm for polynomials, a method for generating cyclic codes, matrix description of cyclic codes, Golay codes. | 5 |
| 6 | BCH Codes  
Primitive elements, minimal polynomials, generator polynomials in terms of minimal polynomials, examples of BCH codes. | 5 |
| 7 | Convolutional Codes  
Tree codes, trellis codes, polynomial description of convolutional codes, distance notions for convolutional codes, the generating function, matrix representation of convolutional codes, decoding of convolutional codes, distance and performance bounds for convolutional codes, examples of convolutional codes, Turbo codes, Turbo decoding. | 5 |
| | Total | 31 |

**Text and Reference Books:**


**PEC-AIML702C: Data Visualization**

**Lecture per week: 3   Credits: 3**

**Course Outcomes (COs):**

At the end of the course, the students will be able to:

1. Identify the different data types, visualization types to bring out the insight.
2. Relate the visualization towards the problem based on the dataset to analyze and bring out valuable insight on large dataset.
3. Design visualization dashboard to support the decision making on large scale data.
4. Demonstrate the analysis of large dataset using various visualization techniques and tools.
5. Identify the different attributes and showcasing them in plots. Identify and create various visualizations for geospatial and table data.
6. Create and interpret plots using R/Python.

**Prerequisites:**

1. Data Mining
### Detailed Content:

<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
</table>
| 1      | Introduction to Data Visualization  
Overview of data visualization - Data Abstraction - Analysis: Four Levels for Validation - Task Abstraction - Analysis: Four Levels for Validation | 4 |
| 2      | Visualization Techniques  
Scalar and point techniques Color maps Contouring Height Plots - Vector visualization techniques Vector properties Vector Glyphs Vector Color Coding Stream Objects. | 5 |
| 3      | Visual Analytics  
Visual Variables - Networks and Trees - Map Color and Other Channels - Manipulate View | 4 |
| 4      | Visual Analytics  
Arrange Tables Geo Spatial data Reduce Items and Attributes | 4 |
| 5      | Visualization Tools and Techniques  
Introduction to data visualization tools- Tableau - Visualization using R | 5 |
| 6      | Diverse Types Of Visual Analysis  
Time- Series data visualization Text data visualization Multivariate data visualization and case studies | 4 |
| 7      | Visualization Dashboard Creations  
Dashboard creation using visualization tools for the use cases: Finance-marketing-insurance-healthcare etc., | 4 |

### Text book and Reference books:

1. Tamara Munzer, Visualization Analysis and Design, CRC Press 2014  
2. Alexandru Telea, Data Visualization Principles and Practice, CRC Press 2014  
3. Paul J. Deitel, Harvey Deitel, Java SE8 for Programmers (Deitel Developer Series) 3rd Ed.

### PEC-AIML-702D: Mobile Computing  
**Lecture per week:** 3  
**Credits:** 3

### Course Outcomes (COs):

At the end of the course, the students will be able to:

1. Understand the necessary knowledge of cellular communication, infrastructure-less networks  
2. Analyze TCP, MAC protocols and their technical feasibility  
3. Analyze device independent applications  
4. Acquire knowledge about the basic concepts and principles in mobile computing  
5. Understand techniques involved, in networks  
6. Analyze systems issues for the design and implementation of mobile computing systems.

### Prerequisites:

- Computer Networks

### Detailed Content:
<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Introduction to Personal Communications Services (PCS): PCS Architecture, Mobility management, Networks signaling. Global System for Mobile Communication (GSM) system overview: GSM Architecture, Mobility management, Network signaling.</td>
<td>4</td>
</tr>
<tr>
<td>2</td>
<td>General Packet Radio Services (GPRS): GPRS Architecture, GPRS Network Nodes. Mobile Data Communication: WLANs (Wireless LANs) IEEE 802.11 standard, Mobile IP</td>
<td>4</td>
</tr>
<tr>
<td>4</td>
<td>Third Generation (3G) Mobile Services: Introduction to International Mobile Telecommunications 2000 (IMT 2000) vision, Wideband Code Division Multiple Access (W-CDMA), and CDMA 2000, Quality of services in 3G.</td>
<td>6</td>
</tr>
<tr>
<td>5</td>
<td>Global Mobile Satellite Systems; case studies of the IRIDIUM and GLOBALSTAR systems. Wireless Enterprise Networks: Introduction to Virtual Networks, Blue tooth technology, Blue tooth Protocols.</td>
<td>6</td>
</tr>
<tr>
<td>6</td>
<td>Server-side programming in Java, Pervasive web application architecture, Device independent example application</td>
<td>4</td>
</tr>
</tbody>
</table>

**Text book and Reference books:**


**OEC-AIML-701A: Internet of Things**

**Lecture per week (L): 3**  **Credits: 3**

**Course Outcomes (COs):**

At the end of the course, the students will be able to:

1. **Understand** the vision of IoT from a global context.
2. **Determine** the Market perspective of IoT.
3. **Design** Devices, Gateways and Data Management in IoT.
4. **Building** state of the art architecture in IoT.
5. **Apply** IoT in Industrial and Commercial Building Automation and Real-World Design.
6. **Evaluate** the performance of IoT devices.

**Prerequisites:**
Detailed Content:

<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>M2M to IoT-The Vision-Introduction, From M2M to IoT, M2M towards IoT-the global context, A use case example, Differing Characteristics.</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>Understand IoT Market perspective. M2M to IoT – A Market Perspective– Introduction, Some Definitions, M2M Value Chains, IoT Value Chains, An emerging industrial structure for IoT, The international driven global value chain and global information monopolies. M2M to IoT- An Architectural Overview– Building an architecture, Main design principles and needed capabilities, An IoT architecture outline, standards considerations.</td>
<td>8</td>
</tr>
</tbody>
</table>

Text book and Reference books:


OEC-AIML-701B: Bioinformatics
Lecture per week (L): 3 Credits: 3
Course Outcomes (COs):

At the end of the course, the students will be able to:

1. **Design** basic algorithms used in Pairwise and Multiple alignments.
2. **Understand** the methodologies used for database searching, and determining the accuracies of database search.
3. **Application** of probabilistic model to determine important patterns.
4. **Prediction** of structure from sequence and subsequently testing the accuracy of predicted structures.
5. **Determine** the protein function from sequence through analyzing data.
6. **Analysis** and development of models for better interpretation of biological data to extract knowledge.

Prerequisites:
- Biological Science
- Basics of Data Structure and Programming

Detailed Content:

<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
</table>
| 1      | Introduction to Bioinformatics  
A word on Bioinformatics, Introduction, Branches of Bioinformatics, Aims of Bioinformatics, Scope/research areas of bioinformatics | 5 |
| 2      | Sequence and molecular file form and ats, Introduction, Sequence file formats, Sequence conversion tools, Molecular file formats, Molecular file format conversion Databases in bioinformatics & introduction: - Introduction, Biological databases, Classification schema of biological databases, Biological database retrieval system Biological sequence databases, National Centre for biotechnology information (NCBI), Introduction, Tools and databases of NCBI, Database retrieval tool, Sequence submission to NCBI, BLAST, PSI-BLAST, RPS-BLAST, Specialized tools, Nucleotide database, Literature database, Protein database, Gene expression database, GEO, Structural database, Chemical database, Other databases, EMBL Nucleotide Sequence Database, Introduction, Sequence retrieval, Sequence submission at EMBL, Resources of EMBL, Biological annotation and data curation, Sequence. | 8 |
| 3      | Introduction, Concept of alignment, Scoring Matrices, PAM, BLOSUM, Alignment of pairs of sequences, Alignment algorithms, Heuristic methods, Multiple sequence Alignment (MSA) Gene prediction methods: principles and challenges, Introduction, Biological overview, What is gene prediction? Computational methods of gene prediction, Combination of two methods, Why is gene prediction difficult. | 7 |
| 4      | Molecular Phylogeny Introduction, Phenotypic phylogeny and molecular phylogeny, Molecular clocks, Methods of phylogeny, Statistical Evaluation of the obtained phylogenetic trees or validation methods, Software for phylogenetic analysis, Reliability of molecular phylogenetic prediction Molecular Viewers, Introduction, A few molecular viewers, RasMol, Deep view- The Swiss-PDB viewer (SPDBV), Cn3D. | 5 |
Text book and Reference books:

1. Bioinformatics: A Textbook, Wiley Online Library

OEC-AIML-701C: Robotics
Lecture per week (L): 3 Credits: 3

Course Outcomes (COs):

At the end of the course, the students will be able to:
1. Perform kinematic and dynamic analyses with simulation.
2. Design control laws for a robot.
3. Apply sensor and vision system for controlling a robot.
4. Integrate mechanical and electrical hardware for a real prototype of robotic device.
5. Develop mathematical model to represent dynamic system.
6. Select a robotic system for given application.

Prerequisites:
- Basic Engineering Mathematics

Detailed Content:

<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/ Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Introduction to Robotics: Types and components of a robot, Classification of robots, closed-loop and open loop control systems, Kinematics systems; Definition of mechanisms and manipulators, Social issues and safety.</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>Robot Kinematics and Dynamics: Kinematic Modelling: Translation and Rotation Representation, Coordinate transformation, DH parameters, Jacobian, Singularity, and Statics, Dynamic Modelling: Equations of motion: Euler-Lagrange formulation.</td>
<td>6</td>
</tr>
<tr>
<td>3</td>
<td>Sensors and Vision System: Sensor: Contact and Proximity, Position, Velocity, Force, Tactile etc. Introduction to Cameras, Camera calibration, Geometry of Image formation, Euclidean/Similarity/Affine/Projective transformations. Vision applications in robotics.</td>
<td>9</td>
</tr>
<tr>
<td>4</td>
<td>Robot Control: Basics of control: Transfer functions, Control laws: P, PD, PID Non-linear and advanced controls</td>
<td>10</td>
</tr>
</tbody>
</table>
Text book and Reference books:


OEC-AIML-701D: Compiler Design

Lecture per week (L): 3 Credits: 3

Course Outcomes (COs):

At the end of the course, the students will be able to:
1. Describe the compilation phases, input and output of each phases, recognizing different possible errors detected by different phases.
2. Design a DFA to recognize partial keywords of C programming language and lexical analyzer, automated lexical analyzer using LEX, and FLEX.
3. Understand the role of a parser, syntax analysis phase, top-down parsing and bottom-up parsing techniques and automated parser generation tool: YACC.
4. Realize and explain the role of semantic analysis phase with data type handling issues, associated actions with CFG rules using SDD and SDT.
5. Understand different ways of intermediate code generation techniques and intermediate representation of any high-level language code segment, and run-time environment issues during compilation.
6. Develop the knowledge of code optimization and code generation issues, construct flow graphs and DAG representation of basic blocks, and register allocation and assignment.

Prerequisites:

1. Automata Theory
2. Knowledge of Programming Language

Detailed Content:

<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Introduction to Compiling Compilers, Analysis of the source program, The phases of the compiler, Cousins of the compiler.</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>Lexical Analysis</td>
<td>4</td>
</tr>
<tr>
<td>3</td>
<td>Syntax Analysis</td>
<td>The role of a parser, Context free grammars, Writing a grammar, Top down Parsing, Non-recursive Predictive parsing (LL), Bottom up parsing, Handles, Viable prefixes, Operator precedence parsing, LR parsers (SLR, LALR), Parser generators (YACC). Error Recovery strategies for different parsing techniques.</td>
</tr>
<tr>
<td>4</td>
<td>Syntax directed translation</td>
<td>Syntax director definitions, Construction of syntax trees, Bottom-up evaluation of S attributed definitions, L attributed definitions, Bottom-up evaluation of inherited attributes.</td>
</tr>
<tr>
<td>5</td>
<td>Type checking</td>
<td>Type systems, Specification of a simple type checker, Equivalence of type expressions, Type conversions.</td>
</tr>
<tr>
<td>6</td>
<td>Run time environments</td>
<td>Source language issues (Activation trees, Control stack, scope of declaration, Binding of names), Storage organization (Subdivision of run-time memory, Activation records), Storage allocation strategies, Parameter passing (call by value, call by reference, copy restore, call by name), Symbol tables, dynamic storage allocation techniques.</td>
</tr>
<tr>
<td>7</td>
<td>Intermediate code generation</td>
<td>Intermediate languages, Graphical representation, Three-address code, Implementation of three address statements (Quadruples, Triples, Indirect triples).</td>
</tr>
<tr>
<td>9</td>
<td>Code generations</td>
<td>Issues in the design of code generator, a simple code generator, Register allocation &amp; assignment.</td>
</tr>
</tbody>
</table>

**Text book and Reference books:**

2. Holub - “Compiler Design in C” - PHI.

**HSMC-AIML-701: Principles of Management**

**Contacts:** 3L per week  
**Credits:** 3

**Course Outcomes:**

At the end of this course, students will demonstrate the ability to
• Learn the basic concept of management, function of management, planning, organizational structure, and organizational effectiveness.
• Understand in depth relationship of management and society.
• Express knowledge of people management & managerial competency.
• Understand leadership theory, decision making process & knowledge development in economic, financial and quantitative analysis.
• Learn about market and customer management.
• Demonstrate in-depth knowledge of operation, technology management and different quality assurance drivers in the industry and their practical usage.

Detailed Content:

<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>Management and Society – Concept, External Environment, CSR, Corporate Governance, Ethical Standards. People Management – Overview, Job design, Recruitment &amp; Selection, Training &amp; Development, Stress Management. Managerial Competencies – Communication, Motivation, Team Effectiveness, Conflict Management, Creativity, Entrepreneurship.</td>
<td>8</td>
</tr>
</tbody>
</table>

Text book and Reference books:

---

**Semester-VIII**

PEC-AIML-801A: Natural Language Processing
Contacts: 3L per week Credits: 3

Course Outcomes (COs):

Upon completion of the course, the students will be able to:
1. Justify the various steps necessary for processing natural language
2. Suggest appropriate lexical and parsing techniques for a given natural language
3. Apply appropriate statistical models for a given natural language application
4. Modify existing algorithms to suit any natural language for processing
5. Recommend appropriate pre-processing steps essential for the various applications involving natural language processing.
6. Design applications involving natural language

**Prerequisites:**
1. Linear algebra
2. Probability and Statistics
3. Artificial Intelligence and Neural Networks
4. Programming in any high level language, preferably python

**Detailed Content:**

<table>
<thead>
<tr>
<th>Unit</th>
<th>Content</th>
<th>Hrs/Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Regular Expressions and Automata (Recap) [2L]</td>
<td>11</td>
</tr>
<tr>
<td></td>
<td>Introduction to NLP, Regular Expression, Finite State Automata</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Tokenization [5L]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Word Tokenization, Normalization, Sentence Segmentation, Named Entity</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Recognition, Multi Word Extraction, Spell Checking - Bayesian Approach,</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Minimum Edit Distance. Morphology [4L]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Morphology - Inflectional and Derivational Morphology, Finite State</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Morphological Parsing, The Lexicon and Morphotactics, Morphological</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Parsing with Finite State Transducers, Orthographic Rules and Finite</td>
<td></td>
</tr>
<tr>
<td></td>
<td>State Transducers, Porter Stemmer</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Language Modeling [4L]</td>
<td>8</td>
</tr>
<tr>
<td></td>
<td>Introduction to N-grams, Chain Rule, Smoothing - Add-One Smoothing,</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Witten-Bell Discounting; Backoff, Deleted Interpolation, N-grams for</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Spelling and Word Prediction, Evaluation of language models.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Hidden Markov Models and POS Tagging [4L]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Markov Chain, Hidden Markov Models, Forward Algorithm, Viterbi</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Algorithm, Part of Speech Tagging - Rule based and Machine Learning</td>
<td></td>
</tr>
<tr>
<td></td>
<td>based approaches, Evaluation</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Text Classification [4L]</td>
<td>9</td>
</tr>
<tr>
<td></td>
<td>Text Classification, Naïve Bayes’ Text Classification, Evaluation,</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Sentiment Analysis - Opinion Mining and Emotion Analysis, Resources</td>
<td></td>
</tr>
<tr>
<td></td>
<td>and Techniques Context Free Grammar [5L]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Context Free Grammar and Constituency, Some common CFG phenomena for</td>
<td></td>
</tr>
<tr>
<td></td>
<td>English, Top-Down and Bottom-up parsing, Probabilistic Context Free</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Grammar, Dependency Parsing</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Computational Lexical Semantics [4L]</td>
<td>9</td>
</tr>
<tr>
<td></td>
<td>Introduction to Lexical Semantics - Homonymy, Polysemy, Synonymy,</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Thesaurus - WordNet, Computational Lexical Semantics - Thesaurus based</td>
<td></td>
</tr>
<tr>
<td></td>
<td>and Distributional Word Similarity. Information Retrieval [5L]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Boolean Retrieval, Term-document incidence, The Inverted Index, Query</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Optimization, Phrase Queries, Ranked Retrieval - Term Frequency -</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Inverse Document Frequency based ranking, Zone Indexing, Query term</td>
<td></td>
</tr>
<tr>
<td></td>
<td>proximity, Cosine ranking, Combining different features for ranking,</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Search Engine Evaluation, Relevance Feedback</td>
<td></td>
</tr>
</tbody>
</table>

**Text Books and References:**
1. Speech and Language Processing, Jurafsky and Martin, Pearson Education
PEC-AIML-801B: Ethical Issues in AI
Contacts: 3L per week Credits: 3

Course Outcomes (COs):

At the end of the course, a student should be able to:
1. Describe knowledge of philosophical issues involved in ethics of artificial intelligence
2. Demonstrate familiarity with relevant examples of AI systems
3. Value the ability to work in a small team
4. Develop written work regularly to a deadline
5. Acquire ability to express arguments clearly and concisely
6. Construct skills in research, analysis and argumentation

Prerequisites:
1. Knowledge about Artificial Intelligence

Detailed Content:

<table>
<thead>
<tr>
<th>Unit</th>
<th>Contents</th>
<th>Hrs/Unit</th>
</tr>
</thead>
</table>
| 1    | AI, information transmission, information processing, and privacy:  
|      | o Big data and privacy  
|      | o Big data and human identity  
|      | o Gender and cultural bias | 6 |
| 2    | Ethics of information and Ethics of AI  
|      | o Ethical issues for different strengths/grades of AI and AI algorithms  
|      | o Medium to strong AI: the moral relevance and effects of its ontological differences | 5 |
| 3    | Normative ethics proposals: advantages and disadvantages:  
|      | o Rule consequentialism  
|      | o Deontological approaches  
|      | o Care ethics  
|      | o Virtue Ethics  
|      | o Problems with implementation  
|      | o Problems with uptake and enforcement | 9 |
| 4    | Software qualities and normative ethics  
|      | o Interpretability, transparency and normative ethics  
|      | o Interpretability, transparency and policy making  
|      | o Extensibility, usability, and communicability | 8 |
| 5    | Ethics of AI on the Web and in Web based applications  
|      | The relationship between AI and the posthuman | 4 |
| 6    | Strong AIs as potential epistemic and moral agents  
|      | o Models, representations, and introspection  
|      | o Interventions and counterfactuals  
|      | o Emulating plasticity: synaptic plasticity versus intrinsic plasticity  
|      | o Imagination | 8 |

Text Books/ Reference Books:
PEC-AIML-801C: Digital Image Processing

Contacts: 3L per week Credits: 3

Course Outcomes (COs):

At the end of the course, the students will be able to:

- Review the fundamental concepts of a digital image processing system.
- Analyze images in the frequency domain using various transforms.
- Evaluate the techniques for image enhancement and image restoration.
- Categorize various compression techniques.
- Interpret Image compression standards.
- Interpret image segmentation and representation techniques.

Prerequisites:

- Digital Signal Processing
- Linear Algebra
- Signals and Systems
- Discrete Mathematics

Detailed Content:

<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td><strong>Introduction:</strong> Overview of Image Processing, Application area of image processing, Digital Image Representation, Sampling &amp; quantization. Spatial and Intensity resolution, interpolation, Relationship between pixels – Neighbors, Adjacency, connectivity, Regions, Boundaries and Distance.</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td><strong>Image Enhancement in Spatial Domain:</strong> Image Quality and Need for image enhancement, Intensity transformation – negative, log, power-law and contrast stretching (linear and non-linear) Histogram based techniques, Spatial Filtering concepts, Spatial Convolution and Correlation, Image smoothing and Sharpening spatial filters. <strong>Image Enhancement in Frequency Domain:</strong> Properties of 1-D and 2-D Discrete Fourier Transform (DFT), Basic of filtering in the frequency domain. Image smoothing and sharpening in frequency domain.</td>
<td>9</td>
</tr>
<tr>
<td>3</td>
<td><strong>Image Restoration:</strong> Introduction to degradation, Types of Image degradations, image degradation models, noise modeling, Estimation of degradation functions, Image restoration in presence of noise only – spatial filtering, Periodic noise and band – pass and band reject filtering.</td>
<td>5</td>
</tr>
<tr>
<td>4</td>
<td><strong>Image Compression:</strong> coding redundancy, Image compression model, Compression Methods – Huffman coding, Arithmetic coding, LZW coding, Run-length coding, and Predicative coding and Vector quantization.</td>
<td>6</td>
</tr>
</tbody>
</table>

---

**Text book and Reference books:**

5. Bhabatosh Chanda, Dwijesh Dutta Majumder, Digital Image Processing and Analysis, Prentice Hall of India

---

**PEC-AIML-801D: Applications of AI in Biomedical Engineering**

**Lecture per week (L): 3 Credits: 3**

**Course Outcomes (COs):**

At the end of the course, the students will be able to:

1. Understand models of human and artificial intelligence, specifically computational models of intelligence.
2. Comprehend a collection of machines learning models (identified and covered in the course), and their applications in medicine and healthcare.
3. Identify and apply appropriate intelligent system models and computational tools to specific problems in biomedicine and healthcare.
4. Analyze the performance of specific models as applied to biomedical problems, and justify their use and limitations.
5. Identify, understand, and interpret methods and evidence from artificial intelligence and other relevant literature.
6. Effectively communicate and disseminate knowledge in any science or engineering domain in the context of computing, systems, and/or biomedical applications.

**Prerequisites:**

1. Biology
2. Probability & Statistics

**Detailed Content:**

<table>
<thead>
<tr>
<th>Sl No</th>
<th>Content</th>
</tr>
</thead>
</table>
| 1     | Foundations  
|       | 1. Introduction to Human and Artificial Intelligence: terminologies, computational models of intelligence; conceptual frameworks from cognitive and educational psychology, neuroscience, information theory, and linguistics; philosophical foundations of AI  
|       | 2. Review of relevant mathematical and statistical concepts: logarithmic loss, cross entropy optimizing cost functions; linear and logistic regression. | 10 |
3. Forms of Learning: supervised, semi-supervised, unsupervised, active, and transfer learning
5. Unsupervised Learning: basic and advanced clustering techniques, dimensionality reduction (feature selection and feature extraction)
6. Knowledge Representation and Reasoning: Prepositional logic, first-order logic, ontological engineering, probabilistic reasoning
7. Time-series analysis: temporal models (probabilistic reasoning over time)
8. Emerging paradigms and concepts in artificial social and emotional intelligence

<table>
<thead>
<tr>
<th>2</th>
<th>Applications</th>
</tr>
</thead>
<tbody>
<tr>
<td>9. Unique characteristics and challenges in medicine and healthcare; History and status quo of intelligent and expert systems in medicine.</td>
<td></td>
</tr>
<tr>
<td>10. Risk stratification, patient outcome prediction, disease progression modelling</td>
<td></td>
</tr>
<tr>
<td>11. Clinical decision-making and intelligent systems to support evidence-based medicine</td>
<td></td>
</tr>
<tr>
<td>12. Phenotype and clinical/bio-marker discovery, Relevance to personalized medicine</td>
<td></td>
</tr>
<tr>
<td>13. Analysis of tissue morphology and other medical imaging applications</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>3</th>
<th>Implementation and Evaluation</th>
</tr>
</thead>
<tbody>
<tr>
<td>14. Tools and Technologies for implementing AI methods</td>
<td></td>
</tr>
<tr>
<td>15. Model evaluation and performance metrics, cross-validation, model interpretability</td>
<td></td>
</tr>
<tr>
<td>16. Ethics of AI: bias, fairness, accountability, and transparency in machine learning; Ethical, Legal, and Social Issues of AI in medicine and healthcare</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>TOTAL</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>30</td>
<td></td>
</tr>
</tbody>
</table>

**Text/ Reference Books:**


**OEC-AIML-801A: Operation Research**

**Lecture per week (L): 3 Credits: 3**

**Course Outcomes (COs):**

At the end of the course, the students will be able to:

1. Design linear programming tools for optimal utilization of resources in various types of industries.
2. Evaluate transportation problems to minimize cost and understand the principles of assignment of jobs and recruitment policies.
3. Analyze decision making under certainty, uncertainty and conflicting situations.
4. Apply forecasting methods for predicting demands.
5. Understand the basic elements of a Queuing model.
6. Remember and Define PERT/CPM technique for project scheduling and resource allocation in an optimal way.
Prerequisites:

1. Probability and Statistics
2. Discrete Mathematics

Detailed Content:

<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>Transportation Problem: Introduction, Formulation of Transportation Problem (TP), Transportation Algorithm (MODI Method), the Initial Basic Feasible Solution, Moving Towards Optimality. Assignment Problem: Introduction, Mathematical Formulation of the Problem, Hungarian Method Algorithm, Travelling Salesman Problem.</td>
<td>5</td>
</tr>
<tr>
<td>5</td>
<td>Queuing Theory: Basis of Queuing theory, elements of queuing theory, Operating characteristics of a queuing system, Queue discipline, Service Mechanism, Classification of Queuing models, [M/M/1]:{FCFS} Queue System, numerical. Inventory Management: Inventory classification, Different costs associated to Inventory, Inventory models with deterministic demands (EOQ, EPQ and price discount models), inventory classification systems.</td>
<td>6</td>
</tr>
<tr>
<td>6</td>
<td>Job Sequencing: Introduction to sequencing and scheduling models: n job two machines problem, n job 3 machines problem. Decision Theory: Introduction, Decision under certainty, Decision under risk, Decision under uncertainty: Laplace criterion, MaxiMin criterion, MiniMax criterion, savage MiniMax regret criterion, Hurwicz criterion, Decision tree.</td>
<td>4</td>
</tr>
<tr>
<td>7</td>
<td>Replacement Theory: Introduction, Replacement of capital equipment which depreciated with time, replacement by alternative equipment, Group and individual replacement policy.</td>
<td>3</td>
</tr>
</tbody>
</table>

Text book and Reference books:

**OEC-AIML-801B: Economic Policies in India**

**Lecture per week (L): 3  Credits: 3**

**Course Outcomes (COs):**

At the end of the course, the students will be able to:
1. Assemble ideas of the basic characteristics of Indian economy, its potential on natural resources.
2. Evaluate land reforms and green revolutions.
4. Interpret the importance, cause and impact of population growth and its distribution, translate and relate them with economic development.
5. Understand economic planning and developmental issues.
6. Remember WTO and Indian agriculture issues.

**Prerequisites:**
- Economics for Engineers

**Detailed Content:**

<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Economic Development and its Determinants: Approaches to economic development and its measurement, sustainable development; Role of State, market and other institutions; Indicators of development – PQLI, Human Development Index (HDI), gender development indices. Planning in India: Objective and strategy of planning; Failures and achievements of Plans; Developing grass-root organizations for development – Panchayats, NGOs and pressure groups.</td>
<td>8</td>
</tr>
<tr>
<td>2</td>
<td>Demographic Features, Poverty and Inequality: Broad demographic features of Indian population; rural-urban migration; Urbanization and civic amenities; Poverty and Inequality. Resource Base and Infrastructure: Energy; social infrastructure – education and health; Environment; Regional imbalance; Issues and policies in financing</td>
<td>9</td>
</tr>
</tbody>
</table>
Infrastructure development.
The Agricultural Sector: Institutional Structure – land reforms in India; Technological change in agriculture – pricing of agricultural inputs and output; industry; Agricultural finance policy; Agricultural Marketing and Warehousing; Issues Terms of trade between agriculture and in food security – policies for sustainable agriculture.

3 Industrial policy; Public Sector enterprises and their performance; Problem of sick units in India; Privatization and disinvestment debate; Growth and pattern of industrialization; Small-scale sector; Productivity in industrial sector; Exit policy – issues in labour market reforms; approaches for employment generation.

Public Finances: Fiscal federalism – Centre-State financial relations; Finances of central government; Finances of state governments; Parallel economy; Problems relating to fiscal policy: Fiscal sector reforms in India.

4 Money, Banking and Prices: Analysis of price behaviour in India; Financial sector reforms; Interest rate policy; Review of monetary policy of RBI; Money and capital markets; Working of SEBI in India.

External Sector: Structure and direction of foreign trade; Balance of payments; Issues in export-import policy and FEMA; Exchange rate policy; Foreign capital and MNCs in India; The progress of trade reforms in India.

Economic Reforms: Rationale of internal and external reforms; Globalization of Indian economy; WTO and its impact on the different sectors of the economy; Need for and issues in good governance; Issues in competition and safety nets in Indian economy.

Text book and Reference books:


OEC-AIML-801C: Microelectronics and VLSI
Lecture per week (L): 3 Credits: 3

Course Outcomes (COs):

At the end of the course, the students will be able to:

1. Have a basic understanding of the MOS device physics, its working as well as its secondary effects.
2. Design a CMOS circuit for any logic function.
3. Design circuits using alternate logic styles
4. Design and analyze the circuit parameters like delay and power.
5. Draw the layout for combinational logic circuits manually and using tools.
6. Use modern simulation tools to verify the functionality of a circuit.

Prerequisites:
- Knowledge about MOS, MOS-Characteristics, MOS Capacitors, Short Channel MOS, CMOS inverters, MOS Gates.
- Basic knowledge on Analog Circuit Theory and Digital Circuits.

Detailed Content:

<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Introduction to VLSI Design: VLSI Design Concepts, Moor's Law, Scale of Integration (SSI, MSI, LSI, VLSI, ULSI – basic idea only), Types of VLSI Chips (Analog &amp; Digital VLSI chips, General purpose, ASIC, PLA, FPGA) , Design principles (Digital VLSI – Concept of Regularity, Granularity etc), Design Domains (Behavioral, Structural)</td>
<td>6</td>
</tr>
<tr>
<td>2</td>
<td>Micro-electronic Processes for VLSI Fabrication: Silicon Semiconductor Technology- An Overview, Wafer processing, Oxidation, Epitaxial deposition, Ion-implantation &amp; Diffusion, Cleaning, Etching, Photolithography – Positive &amp; Negative photo-resist; Basic CMOS Technology – (Steps in fabricating CMOS), Basic n-well CMOS process, p-well CMOS process, Twin tub process, Silicon on insulator; Layout Design Rule: Stick diagram with examples, Layout rules</td>
<td>10</td>
</tr>
<tr>
<td>3</td>
<td>CMOS for Digital VLSI Circuits: Recapitulation of MOS; CMOS, CMOS inverter characteristics; CMOS logic circuits, NAND &amp; NOR Gates, Complex logic circuits, CMOS Full Adder, CMOS Transmission GATE, Advanced CMOS Logic circuits; Sequential CMOS logic circuits; SR Latch circuit, clocked JK Latch/ Master-Slave JK, CMOS D-latch &amp; Edge triggered flip-flop</td>
<td>10</td>
</tr>
<tr>
<td>4</td>
<td>Analog VLSI Circuits: Analog VLSI design steps; Basic building blocks of Analog VLSI chips; MOS switch; Active load / resistors; Voltage dividers; CMOS Current source &amp; sink; CMOS Voltage references/voltage dividers [Basic circuits only]; CMOS Differential amplifier; Output amplifiers [Basic circuits only]; CMOS OPAMP ; Switched capacitor filter</td>
<td>8</td>
</tr>
</tbody>
</table>

Text book and Reference books:
2. CMOS Digital Integrated Circuit, S.M.Kang & Y.Leblebici, TMH.
4. VHDL, Bhaskar, PHI.
5. Advance Digital Design Using Verilog , Michel D. Celliti, PHI
OEC-AIML-801D: Quantum Computing

Lecture per week (L): 3  Credits: 3

Course Outcomes (COs):

At the end of the course, the students will be able to:

1. Explain the working of a Quantum Computing program, its architecture and program model.
2. Develop quantum logic gate circuits.
3. Design different mathematical foundation for quantum computing.
4. Develop quantum computing algorithm.
5. Program quantum algorithm on major toolkits.
6. Find different quantum computational complexity.

Prerequisites:
- Data Structure and Algorithm
- Programming in Python/C#

Detailed Content:

<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Introduction to Quantum Computing: Motivation for studying Quantum Computing Major players in the industry (IBM, Microsoft, Rigetti, D-Wave etc.), Origin of Quantum Computing Overview of major concepts in Quantum Computing: Qubits and multi-qubits states, Bracket notation. Bloch Sphere representation, Quantum Superposition, Quantum Entanglement</td>
<td>3</td>
</tr>
<tr>
<td>2</td>
<td>Math Foundation for Quantum Computing: Matrix Algebra: basis vectors and orthogonality, inner product and Hilbert spaces, matrices and tensors, unitary operators and projectors, Dirac notation, Eigen values and Eigen vectors</td>
<td>8</td>
</tr>
<tr>
<td>3</td>
<td>Building Blocks for Quantum Program: Architecture of a Quantum Computing platform Details of q-bit system of information representation: Block Sphere, Multi-qubits States, Quantum superposition of qubits (valid and invalid superposition), Quantum Entanglement, Useful states from quantum algorithmic perceptive e.g. Bell State, Operation on qubits: Measuring and transforming using gates. Quantum Logic gates and Circuit: Pauli, Hadamard, phase shift, controlled gates, Ising, Deutsch, swap etc. Programming model for a Quantum Computing Program: Steps performed on classical computer, Steps performed on Quantum Computer, Moving data between bits and qubits.</td>
<td>8</td>
</tr>
<tr>
<td>4</td>
<td>Quantum Algorithms: Basic techniques exploited by quantum algorithms: Amplitude amplification,</td>
<td>10</td>
</tr>
</tbody>
</table>
Quantum Fourier Transform, Phase Kick-back, Quantum Phase estimation, Quantum Walks.  
Major Algorithms: Shor’s Algorithm, Grover’s Algorithm, Deutsch’s Algorithm, Deutsch-Jozsa Algorithm. 
OSS Toolkits for implementing Quantum program: IBM quantum experience Microsoft Q: Rigetti PyQuil (QPU/QVM)  

<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>Quantum Computational Complexity and Error Correction: Computational complexity, Black-box model, Lower bounds for searching general, Black-box lower bounds, Polynomial method, Block sensitivity, Adversary methods, Classical error correction, Classical three-bit code, Fault tolerance, Quantum error correction Three- and nine-qubit quantum codes, Fault-tolerant quantum computation.</td>
<td>7</td>
</tr>
</tbody>
</table>

Text book and Reference books:  
3. IBM Experience:  
   https://quantumexperience.ng.bluemix.net  
4. Microsoft Quantum Development Kit:  
5. Forest SDK PyQuil:  

OEC-AIML-802A: Organizational Behaviour  
Lecture per week (L): 3    Credits: 3

Course Outcomes (COs):

At the end of the course, the students will be able to:
1. **Discuss** the development of the field of organizational behaviour and explain the micro and macro approaches  
2. **Analyze** and compare different models used to explain individual behaviour related to motivation and rewards  
3. **Identify** the processes used in developing communication and resolving conflicts  
4. **Explain** group dynamics and demonstrate skills required for working in groups (team building)  
5. **Evaluate** the various leadership styles and the role of leaders in a decision making process.  
6. **Design** the implementation of organizational change

Prerequisites:  
- Soft Skill and Interpersonal Communication

Detailed Content:

<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Definition, need and importance of organizational behaviour – Nature and scope –</td>
<td>5</td>
</tr>
<tr>
<td>Framework</td>
<td>Organizational behaviour models.</td>
<td></td>
</tr>
<tr>
<td>-----------</td>
<td>----------------------------------</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Individual Behaviour:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Misbehaviour – Types – Management Intervention.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Emotions - Emotional Labour – Emotional Intelligence – Theories.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Motivation – importance – Types – Effects on work behavior.</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Group Behaviour:</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Organization structure – Formation – Groups in organizations – Influence – Group dynamics – Emergence of informal leaders and working norms – Group decision making techniques – Team building- Interpersonal relations – Communication – Control.</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>Leadership and Power:</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>Dynamics of Organizational Behaviour</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Organizational culture and climate – Factors affecting organizational climate – Importance.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Organizational change – Importance – Stability Vs Change – Proactive Vs Reaction change – the change process – Resistance to change – Managing change.</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Organizational development – Characteristics – objectives –. Organizational effectiveness.</td>
<td></td>
</tr>
<tr>
<td>Text book and Reference books:</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**OEC-AIML-802B: Human Resource Management**

**Lecture per week (L): 3  Credits: 3**

**Course Outcomes (COs):**

At the end of the course, the students will be able to:

1. **Demonstrate** an understanding of key terms, theories/concepts and practices within the field of HRM
2. **Design** competence in development and problem-solving in the area of HRM
3. **Formulate** innovative solutions to problems in the fields of HRM
4. Identify and appreciate the significance of the ethical issues in HR
5. Analyze the problem and issues related to human resources in an organization.
6. Integrate the understanding of various HR concepts along with the domain concept in order to take correct business decisions.

Prerequisites:
1. Interpersonal and soft skill
2. English communication

Detailed Content:

<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>HRD-Macro Perspective: HRD Concept, Origin and Need, HRD as a Total System; Approaches to HRD; Human Development and HRD; HRD at Macro and Micro Climate</td>
<td>5</td>
</tr>
<tr>
<td>2</td>
<td>HRD–Micro Perspective: Areas of HRD; HRD Interventions Performance Appraisal, Potential Appraisal, Feedback and Performance Coaching, Training, Career Planning, OD or Systems Development, Rewards, Employee Welfare and Quality of Work Life and Human Resource Information; Staffing for HRD: Roles of HR Developer; Physical and Financial Resources for HRD; HR Accounting; HRD Audit, Strategic HRD</td>
<td>7</td>
</tr>
<tr>
<td>3</td>
<td>Instructional Technology for HRD Learning and HRD; Models and Curriculum; Principles of Learning; Group and Individual Learning; Transactional Analysis; Assessment Centre; Behaviour Modeling and Self Directed Learning.</td>
<td>7</td>
</tr>
<tr>
<td>4</td>
<td>Human Resource Training and Development: Concept and Importance; Assessing Training Needs; Designing and Evaluating T&amp;D Programmes; Role, Responsibilities and challenges to Training Managers.</td>
<td>5</td>
</tr>
<tr>
<td>5</td>
<td>Training Methods: Training with in Industry (TWI): On the Job &amp; Off the Job Training; Management Development: Lecture Method; Role Play; In-basket Exercise; Simulation; Vestibule Training; Management Games; Case Study; Programmed Instruction; Team Development; Sensitivity Training; Globalization challenges and Strategies of Training; Program, Review on T&amp;D Programmes in India</td>
<td>7</td>
</tr>
</tbody>
</table>

Text book and Reference books:

OEC-AIML-802C: Research Methodology
Lecture per week: 3 Credits: 3

Course Outcomes (COs):
At the end of the course, the students will be able to:

1. Discuss different methodologies and techniques used in research work.
2. Explain basic computer skills necessary for the conduct of research.
3. Explain key research concepts and issues.
4. Select and define appropriate research problem and parameters.
5. Develop the required numerical skills necessary to carry out research.
6. Develop an appropriate framework for research studies.

Prerequisites:
- Probability and Statistics

Detailed Content:

<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Research Formulation and Design: Motivation and objectives – Research methods vs. Methodology. Types of research – Descriptive vs. Analytical, Applied vs. Fundamental, Quantitative vs. Qualitative, Conceptual vs. Empirical, concept of applied and basic research process, criteria of good research. Defining and formulating the research problem, selecting the problem, necessity of defining the problem, importance of literature review in defining a problem, literature review-primary and secondary sources, reviews, monograph, patents, research databases, web as a source, searching the web, critical literature review, identifying gap areas from literature and research database, development of working hypothesis.</td>
<td>8</td>
</tr>
<tr>
<td>2</td>
<td>Data Collection and Analysis: Accepts of method validation, observation and collection of data, methods of data collection, sampling methods, data processing and analysis strategies and tools, data analysis with statically package (Sigma STAT, SPSS for student t-test, ANOVA, etc.), hypothesis testing</td>
<td>8</td>
</tr>
<tr>
<td>3</td>
<td>Research Ethics, IPR and Scholarly Publishing: Ethics-ethical issues, ethical committees (human &amp; animal); IPR- intellectual property rights and patent law, commercialization, copy right, royalty, trade related aspects of intellectual property rights (TRIPS); scholarly publishing- IMRAD concept and design of research paper, citation and acknowledgement, plagiarism, reproducibility and accountability.</td>
<td>8</td>
</tr>
<tr>
<td>4</td>
<td>Interpretation and Report Writing: Meaning of Interpretation, Technique of Interpretation, Precaution in Interpretation, Significance of Report Writing, Different Steps in Writing Project Report, Layout of the Project/Research Report, Types of Reports, Oral Presentation, Mechanics of Writing a Project/Research Report, Precautions for Writing Research Reports, Conclusions.</td>
<td>8</td>
</tr>
</tbody>
</table>

Text book and Reference books:


OEC-AIML-802D: Soft Skill and Interpersonal Communication
Lecture per week: 3 Credits: 3

Course Outcomes (COs):

At the end of the course, the students will be able to:
1. Develop effective communication skills (spoken and written).
2. Develop effective presentation skills.
3. Develop all-round personalities with a mature outlook to function effectively in different circumstances.
4. Develop broad career plans, evaluate the employment market, identify the organizations to get good placement, match the job requirements and skill sets.
5. Improve self-confidence.
6. Become more effective individual through goal/target setting, self-motivation and practicing creative thinking.

Prerequisites:

- English Grammar

Detailed Content:

<table>
<thead>
<tr>
<th>Module</th>
<th>Content</th>
<th>Hours/Module</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>An Introduction – Definition and Significance of Soft Skills; Process, Importance and Measurement of Soft Skill Development. Inter personal relations; communication models, process and barriers; team communication; developing interpersonal relationships through effective communication; listening skills; essential formal writing skills; corporate communication styles –assertion, persuasion, negotiation</td>
<td>6</td>
</tr>
<tr>
<td>2</td>
<td>SWOT &amp; Creative Thinking Discovering the Self; Setting Goals; Beliefs, Values, Attitude, Virtue. Developing Positive Thinking and Attitude; Driving out Negativity; Meaning and Theories of Motivation; Enhancing Motivation Levels.</td>
<td>6</td>
</tr>
<tr>
<td>3</td>
<td>Corporate Communication Public Speaking: Skills, Methods, Strategies and Essential tips for effective public speaking. Group Discussion: Importance, Planning, Elements, Skills assessed; Effectively</td>
<td>6</td>
</tr>
</tbody>
</table>
disagreeing, Initiating, Summarizing and Attaining the Objective. 

| 5 | Business Etiquette & Team Work Concept of Teams; Building effective teams; Concept of Leadership and honing Leadership skills. Meaning, Nature, Features, Stages, Models; Learning Skills; Adaptability Skills |

**Text book and Reference books:**

2. Effective Communication and Soft Skills, Nitin Bhatnagar, Pearson Education India, 2011