Learning Outcomes based Curriculum Framework

(LOCF)

For

B. Tech. Artificial Intelligence & Machine Learning

Year II (Semester 3 & 4)

Batch 2022-23 Onward

Faculty of Engineering and Technology

Chaudhary Devi Lal University Sirsa-125055

Scheme of Examination & Detailed Syllabus of BTech (AI & ML) 2nd Year

(3rd & 4th Semester)

Program Specific Outcomes (PSOs)

PSO1 **Developing Computational Systems**: Use principles of various programming languages, data structures, database management systems, computer algorithms, theory of computation, networking and software engineering for designing and implementing computational systems.

PSO2 **Designing Intelligent Machine Learning Systems:** Utilize the principles and tools of artificial intelligence, soft computing, data mining and machine learning, data analytics, robotics, IoT, augmented reality etc. for designing and working with intelligent systems that learn from their environment.

Programme Outcomes (POs) of Bachelor Programmes in Engineering & Technology have been specified in First Year common curriculum of B.Tech. programmes.

Course Code	**Definition/ Category**
BSC	Basic Science Courses
ESC	Engineering Science Courses
HSMC	Humanities and Social Sciences including Management Courses
MC	Mandatory Audit Courses
PC	Program Core
PE	Program Elective Courses
OE	Open Elective Courses
EEC	Employability Enhancement Courses (Project work/ Summer Training/ Industrial Training/ Practical Training/ Internship/Seminar, etc.)

B.Tech. AI & ML 3rd & 4thSemester

Semester	Basic Sciences' Courses BSC		Engineering Sciences'/Programme Core/Programme Elective/Open Elective Courses ESC/PC/PE/OE		Humanities, Social Sciences, Management Courses HSMC		Mandatory Courses MC		Grand Total Credit
	No. of Courses	Total Credits	No. of Courses	Total Credits	No. of Courses	Total Credits	No. of Courses	Total Credits	
3rd	01	03	07	18	00	00	01	0	21
4th	00	00	10	23	00	00	00	0	23

Courses' codes, titles, and credits (Semester- III)

#	Course Code	Course Title	Workload/Credit			
			Theory	Tutorial	Practical	Total
1	BSC/9-T	Mathematics for Machine Learning	3/3	-/-	-	3/3
2	PC/AIML/1-T	Data Structures and Algorithms	3/3	-/-	-	3/3
3	PC/AIML/2-T	Object Oriented Programming using C++	3/3	-/-	-	3/3
4	PC/AIML/3-T	Data Analytics using R	3/3	-/-	-	3/3
5	*MC/2-T	Environmental Science	3/-	-/-	-	3/-
6	PC/AIML/4-T	Fundamentals of AI & ML	3/3	-/-	-	3/3
7	PC/AIML/1-P	Data Structures and Algorithms using C/C++ Lab	-/-	-/-	4/2	4/2
8	PC/AIML/2-P	Object Oriented Programming using C++ Lab.	-/-	-/-	4/2	4/2
9	PC/AIML/3-P	Data Analytics using R Lab.			4/2	4/2
					Total Credit	30/21

Courses' codes, titles, and credits (Semester- IV)

#	Course Code	Course Title	Workload/Credit			
			Theory	Tutorial	Practical	Total
1	PC/AIML/5-T	Computer Organisation and Architecture	3/3	-/-	-	3/3
2	PC/AIML/6-T	Data Mining Techniques	3/3	-/-	-	3/3
3	PC/AIML/7-T	Operating Systems	3/3	-/-	-	3/3
4	PC/AIML/8-T	Database Management System	3/3	-/-	-	3/3
5	PC/AIML/9-T	Discrete Mathematics	3/3	-/-	-	3/3
6	PC/AIML/10-T	Python Programming	3/3	-/-	-	3/3
7	PC/AIML/6-P	Data Mining using R/Python/WEKA Lab.	-/-	-/-	4/2	4/2
8	PC/AIML/8-P	Database Management System Lab.	-/-	-/-	2/1	2/1
9	PC/AIML/10-P	Python Programming Lab.	-/-	-/-	4/2	4/2
					Total Credit	28/23
1.	**EEC/AIML/1	Industrial Training/ Internship-I	-/-	-/-	-/4	-/4

*Non-credit qualifying mandatory course.

**The students will have to undergo Industrial/Practical Training/ Internship for 4-6 weeks during summer vacations after the examination of 4th semester which will be evaluated in 5th semester.

Note: Students will be allowed to use non-programmable scientific calculators only, however, sharing of calculator should not be permitted.

MATHEMATICS FOR MACHINE LEARNING

General Course Information

Course Code: BSC/9-T Course Credits: 3 Type: Basic Sciences Contact Hours: 3 hours/week Mode: Lectures (L) Examination Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Mathematics I and Mathematics II

About the Course

This is an advanced mathematics course that offers the knowledge of Fourier Series, Fourier Transforms, Functions of Complex Variables. These concepts are essential for students to solve problems in image processing, digital signal processing and other related engineering fields.

Course Outcomes: By the end of the course students will be able to:

CO1.To understand the basic concepts of data science & machine learning Concepts and their application in modern context
CO2. To apply the basic statistical concepts for solving various problems
CO3.To distinguishes between various probability distributions and apply the concepts for the solution of related problems
CO4. To learn the essential tools of matrices and linear algebra including linear transformations, Eigen values, Diagonalisation, orthogonalization and factorization
CO5.To learn mathematical modelling, types of matrices
CO6.To Implement mathematical concepts using real-world data

Course Content

Unit I

Overview of Data Science & Machine Learning: Introduction and history of Data Science, Introduction and history of Machine Learning, Overlap between Data Science, Machine Learning and Artificial Intelligence, Applications of Data Science & Machine Learning in the modern context, Types of data, Basic Statistical Concepts: Scale of Measurements (Nominal, Ordinal, Ratio and Interval), Measures of Location, Measures of Variability/Spread, Measures of Shape.

Case Studies:-Bollywood Dataset, coronary heart disease dataset.

Unit II

Probability Theory: Principle of counting, definitions of probability theory, independent events, mutually exclusive events, collectively exhaustive events, conditional probability, Bayes Theorem, Discrete probability distribution (Discrete Uniform Distribution, Poisson Distribution, Bernoulli Distribution and Binomial Distribution), covariance, correlation, Continuous probability distribution, normal distribution, Central Limit Theorem, Binomial Distribution, Continuous Uniform Distribution, Exponential Distribution, P-Value, T-Value, Confidence Interval, t distribution and chi square distribution

Unit III

Linear Algebra: Introduction to linear algebra, notations and definitions, Elementary transformations, Elementary matrices, inverse using elementary transformations, Rank of a matrix, Normal form of a matrix, Linear dependence and independence of vectors, Consistency of linear system of equations, Eigen Values and Eigen vectors, Properties of Eigen values, Cayley Hamilton theorem, Linear Transformation, Orthogonal transformation

Unit IV

Mathematical modelling: Similar matrices, Diagonalisation of a matrix Operations on matrices - additions, subtraction, multiplication, scalar multiplication, vector multiplication, Orthogonal Matrix, Singularity of Matrix, Matrix factorization, decomposition such as LU, QR and SVD, Conceptualizing a mathematical model/curve form first principle, concept of boundary conditions

Text and Reference Books:

1. Probability for Statistics and Machine Learning:Anirban DasGupta - 2011
2. Probability for Machine Learning: Discover How To Harness, Jason Brownlee – 2019.
3. Machine Learning using Python: Manaranjan pradhan, U Dinesh Kumar-2020, Wiley.
4. Machine Learning using Python, PRADHAN, Manaranjan, 1st and Reprint 2019, Wiley.
5. Machine Learning (in Python and R),) MUELLER,John Paul ,Wiley India
6. Aurélien Géron, "Hands-On Machine Learning with Scikit-Learn and Tensor Flow: Concepts, Tools, and Techniques to Build Intelligent Systems", 1st Edition, O'Reilly Media
7.Python for Probability, Statistics, and Machine Learning, José Unpingco – 2019
8.Mathematics for Machine Learning, Marc peter Deisenroth, A. Aldo Faisal and Cheng Soon Ong. Published by Cambridge University press.
9. Jeeva Jose, “Machine Learning”, Khanna Publishing House, Delhi.
10.Ian Goodfellow, Yoshua Bengio and Aaron Courville, "Deep Learning", MIT Press http://www.deeplearningbook.org

CO-PO Articulation Matrix: Mathematics for Machine Learning

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1.To understand the basic concepts of data science & machine learning Concepts and their application in modern context	1	--	-		-	-	-	-	-	-	-	-	2	2
CO2. To apply the basic statistical concepts for solving various problems	2	2	2	2	-	-	-	-	-	-	-	-	3	2
CO3.To distinguish between various probability distributions and apply the concepts for the solution of related problems	2	2	2	2	-	-	-	-	-	-	-	-	3	2
CO4. To learn the essential tools of matrices and linear algebra including linear transformations, eigen values, diagonalisation, orthogonalization and factorization	3	3	2	3	-	-	-	-	-	-	-	-	3	2
CO5.To learn mathematical modelling, types of matrices	3	3	2	3	-	-	-	-	-	-	-	-	3	2
CO6.To Implement mathematical concepts using real-world data	3	3	2	3	-	-	-	-	-	-	-	-	2	2

DATA STRUCTURES AND ALGORITHMS

General Course Information

Course Code: PC/AIML/1-T Course Credits: 3 Type: Professional Core Contact Hours: 3hours/week Mode: Lectures (L) Examination Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Programming in C

About the Course:

Data Structure and Algorithms is a core and an essential course for every graduate in Computer Science and Engineering. This course introduces data structures like arrays, linked lists, trees and graphs etc. and various operations to be implemented on these data structures for solving real world problems. It includes various sorting and searching algorithms as well. Further, it incorporates complexity analysis of algorithms implemented on various data structures.

Course Outcomes: By the end of the course students will be able to:

CO1. **Describe** various types of data structures and operations that can be implemented on these data structures. (LOTS: Level 1: Remember)

CO2. **Demonstrate** the use of various data structures and their related operations. (LOTS: Level 2: Understand)

CO3. **Apply** data structure to solve computational problems. (LOTS: Level 3: Apply)

CO4. **Compare** the suitability of alternative data structures and prescribed operations for various problem situations. (HOTS: Level 4: Analyse).

CO5. **Defend** solutions with respect to effective storage of data and efficiency of the required operations for solving real world problems. (HOTS: Level 5: Evaluate)

Course Content

Unit I

Introduction to data structures and their types, Abstract data types, linear lists: Arrays and linked lists: memory representations, implementing operations like traversing, searching, inserting and deleting etc. Applications of arrays and linked lists. Representing sets and polynomials using linked lists.

Unit II

Stack and Queue: Static and linked implementations, Operations and Applications. Circular queues, Tress, Binary trees and related terminology, Tree traversals (Recursive), Threaded Binary Trees, Binary Search Trees implementation and operations, Priority queues.

Unit III

Height Balanced or AVL trees and B trees. Graph definitions and related terminology, memory representations and related operations (traversal, insertion, deletion, search), Path Matrix, Warshall's Shortest path algorithm Hashing, Hash tables, hash function and collision resolution.

Unit IV

Sequential and binary search, Sorting algorithms: Bubble sort, Selection sort, Insertion sort, Quick sort, Merge sort, Count sort, Heap sort, Comparison of searching and sorting techniques based on their complexity analysis, Time and space complexity of algorithms: Asymptotic analysis, Big O, Omega, Theta notations.

Text and Reference Books:

1. Aho, A. V., Ullman, J. D., and Hopcroft, J. E., *Data Structures and Algorithms*, Addison-Wesley, 1983.
2. LangsamYedidyah, Augenstein J Moshe, Tenenbaum M Aaron, *Data Structures using C and C++*, 3rdedition, PHI, 2009.
3. Cormen, T. H., Leiserson, C. E., Rivest, R. L. and Stein, C., *Introduction to Algorithms*, MIT Press, 2009.
4. Robert L. Kruse, *Data Structure and Program Design in C*, Pearson Education India, 2007.
5. Weiss, M. A., *Data Structures and Algorithm Analysis in C++*, Addison-Wesley, 2007.
6. Sahni, S., *Data Structures, Algorithms, and Applications in C++*, WCB/McGraw-Hill, 2001.

CO-PO Articulation Matrix: Data Structures and Algorithms (PC/AIML/1-T)

Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
CO1. **Describe** various types of data structures and operations that can be implemented on these data structures. (LOTS: Level 1: Remember)	1	-	-	-	-	-	-	-	-	-	-	-	3	-
CO2. **Demonstrate** the use of various data structures and their related operations. (LOTS: Level 2: Understand)	1	-	-	-	-	-	-	-	-	-	-	-	3	-
CO3. **Apply** data structure to solve computational problems. (LOTS: Level 3: Apply)	2	2	-	-	2	-	-	-	-	-	-	-	3	2
CO4. **Compare** the suitability of alternative data structures and prescribed operations for solving a problem. (HOTS: Level 4: Analyse).	2	2	-	-	-	-	-		-	-	-	-	3	2
CO5. **Defend** solutions with respect to effective storage of data and efficiency of the required operations for solving computational problems. (HOTS: Level 5: -Evaluate)	3	3	-	1	-	-	-	-	-	-	-	-	3	2
Level of Attainments PC/AIML/1-T														

OBJECT ORIENTED PROGRAMMING USING C++

General Course Information

Course Code: PC/AIML/2-T Course Credits: 3 Type: Professional Core Contact Hours: 3 hours/week Mode: Lectures (L) Examination Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Knowledge of computer fundamentals and problem solving using C programming

About the Course:

Objected Oriented Programming using C++ is an essential course for every graduate in Computer Science and Engineering. This course introduces the Object Oriented concept such as data encapsulation, data hiding, data abstraction, reusability, exception handling etc., and their implementation using C++.

Course Outcomes: By the end of the course students will be able to:

CO1. **List** the concepts related to object oriented paradigms. (LOTS: Level 1: Remember)

CO2. **Distinguish** between structured and object oriented approaches to programming. (LOTS: Level 2: Understand)

CO3. **Apply** object oriented constructs for problem solving. (LOTS: Level 3: Apply)

CO4. **Detect** logical and run time errors and suggest appropriate modifications. (HOTS: Level 4: Analyze)

CO5. **Justify** the design of a program for a given problem. (HOTS: Level 5: Evaluate)

CO6. **Design** solutions to programming problems using multiple object oriented programming constructs together. (HOTS: Level 6: Create)

Course Content

Unit I

Introduction to object oriented programming, C++ standard library, basics of a typical C++ environment, illustrative simple C++ programs, new features of ANSI C++ standard, OOPs concepts: Information hiding, encapsulation, data abstraction, access modifiers, controlling access to a class level, method, or variable (public, protected, private, block level, scope and mutable), other modifiers. Structure of class and struct in memory, accessing members of structures, Class scope and accessing class members, separating interface from implementation, pre-processors directives, macro programs, header files and namespaces, default constructors, chained constructor, default arguments with constructors, constant object and const member functions, object as member of class, use of destructors, virtual destructors, controlling access function and utility functions, function overloading.

Unit II

Inline function, friend function and friend classes, using this pointer, dynamic memory allocation with new and delete, static class members, proxy class, polymorphism concepts, overloading, overriding methods, abstract classes, reusability, class's behaviors, inheritance, base classes and derived classes, protected members, casting base-class pointers to derived-class pointers, using member functions, overriding base-class members in a derived-class, public, protected and private inheritance, using constructors and destructors in derived classes, implicit derived-class object to base- class object conversion, composition vs. inheritance.

Unit III

Virtual functions, abstract base classes and concrete classes, new classes and dynamic binding, virtual destructors, fundamentals of operator overloading, restrictions on operators overloading, operator functions as class members vs. as friend functions, overloading, <<, >> overloading unary operators, overloading binary operators. I/O Streams, file handling, creating a sequential access file, reading data from a sequential access file, updating sequential access files, random access files, creating a random access file, writing data randomly to a random access file, reading data sequentially from a random access file.

Unit IV

Managing Console I/O, stream input/output classes and objects, stream output, stream input, unformatted I/O (with read and write), stream manipulators, stream format states, stream error states, exception handling, basics of C++ exception handling(try, throw, catch), Rethrowing an exception, specific exception, processing unexpected exceptions, stack unwinding, exception handling in constructors and destructors, inheritance with exception introduction to generic classes, function templates, overloading template functions, class template, non-type parameters, templates and inheritance, templates and friends, templates and static members, container, iterator, algorithm and functional classes.

Text and Reference Books:

1. H. M.Deitel and P. J.Deitel, *C++ How To Program*, 6th Ed., Prentice Hall, 2008.
2. Robert Lafore, *Object-Oriented Programming in C*++, 3rd Ed., Sams Publishing, 2001.
3. D. Ravichandran, *Programming with C*++, 3rd Ed., T.M.H, 2011.
4. E.Balagurusamy, *Object oriented Programming with C*++, 6th Ed., Tata McGraw-Hill,2013.
5. Horstmann, *Computing Concepts with C++ Essentials*, 3rd Ed., John Wiley,2003.
6. Herbert Schildt , *The Complete Reference in C*++, 5th Ed., TMH, 2012.

CO-PO Articulation Matrix: Object Oriented Programming Using C++ (PC/AIML/2-T)

Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
CO1. **List** the concepts related to object oriented paradigms. (LOTS: Level 1: Remember)	1	1	-	-	-	-	-	-	-	-	-	-	2	-
CO2. **distinguish** between structured and object oriented approaches to programming. (LOTS: Level 2: Understand)	1	1	-	-	-	-	-	-	-	-	-	-	3	-
CO3. **Apply** object oriented constructs for problem solving. (LOTS: Level 3: Apply)	2	1	-	-	2	-	-	-	-	-	-	-	3	-
CO4. **Detect** logical and run time errors and suggest appropriate modifications. (HOTS: Level 4: Analyse)	2	2	-	-	-	-	-	-	-	-	-	-	3	-
CO5. **Justify** the design of a program for a given problem. (HOTS: Level 5: Evaluate)	2	3	-	-	-	-	-	-	1	-	-	-	3	-
CO6. **Design** solutions to programming problems using multiple object oriented programming constructs together. (HOTS: Level 6: Create)	3	3	1	-	2	-	-	-	1	-	-	-	3	-
Level of Attainments PC/AIML/2-T														

DATA ANALYTICS USING R

General Course Information

Course Code: PC/AIML/3-T Course Credits: 3 Type: Professional Core Contact Hours: 3hours/week Mode: Lectures (L) Examination Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Basic programming skills, Probability and Statistics

About the Course:

In this course, the learners will be able to develop expertise in R programming for manipulating, exploring, visualizing, applying descriptive and inferential statistics. In addition, they will learn to implement predictive modelling.

Course Outcomes: By the end of the course students will be able to:

CO1**.** **Define** the basic terms related to data analytics. (LOTS: Level 1: Remember)
CO2. **Describe** data with statistical summaries and plots. (LOTS: Level 2: Understanding)
CO3. **Build** predictive models. (LOTS: Level 3: Apply)
CO4. **Analyze** the quality of a statistical and machine learning models. (HOTS: Level 4: Analyse)
CO5. **Interpret** and evaluate statistical and predictive models. (HOTS: Level 5: Evaluate)
CO6. **Conclude** the findings of predictive modelling. (HOTS: Level 5: Evaluate)

Course Content

Unit I

Introduction to R programming: Data types or objects in R, Creating and manipulating objects like factors, vectors and matrices, lists and data frames, Subsetting matrices and data frames, Vectorized operations for vectors and matrices and data frames.

Unit II

Control structure in R: If-else statements, for and while loops, loop functions like lapply, apply, sapply and mapply etc.; writing user defined functions in R. Getting data in and out of R.

Unit III

Doing basic descriptive statistics: Data types for data analysis and their mapping to R objects, Mean, Median, Mode, Quantiles, Five-point summary, Variance, Correlation and Covariance, normal distribution, uniform distribution using R, Hypothesis testing: Chi-Square test and student's T test.

Unit IV

Exploratory Data Analysis: Visualizing data through various plots and charts (bar charts, histogram, frequency polygon, scatter plot, box plots etc.), Applying KNN and Bayesian predictive models.

Text and Reference Books:

1. Hadley Wickham and Garrett Grolemund., *R for Data Science Import, Tidy, Transform and model Data*, O'Reilly, 2017.

2. Roger D. Peng, *R Programming for Data Science*, Lean Publishing, 2015.

3. Paul Teeter, *R Cookbook*, O'Reilly, 2011.

4. W. N. Venables, D. M. Smith and the R core Team, *An introduction to R, Notes on R: A Programming Environment for Data Analysis and Graphics*, version 3.3.2, 2016.

5. Michael J. Crawley, *Statistics, An introduction using R*, Second edition, John Wiley, 2015

6. Han, J., Kamber, M, Pei, J., *Data Mining Concepts and Techniques*, Third edition, Morgan Kaufmann, 2012.

7. Trevor Hastie, Robert Tibshirani, Jerome Friedman, *The Elements of Statistical Learning: Data Mining, Inference and Prediction*, Springer, 2nd edition, 2009

CO-PO Articulation Matrix: Data Analytics using R (PC/AIML/3-T)

Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
CO1. **Define** the basic terms related to data analytics. (LOTS: Level 1: Remember)	1	-	-	-	1	-	-	-	-	-	-	-	-	2
CO2. **Describe** data with statistical summaries and plots. (LOTS: Level 2: Understanding)	2	2	-	-	2	-	-	-	-	-	-	-	1	3
CO3. **Build** predictive models (LOTS: Level 3: Apply).	2	2	-	-	3	-	-	-	-	-	-	-	1	3
CO4. **Analyse** the quality of a statistical and machine learning models (HOTS: Level 4: Analyse).	2	2	-	-	3	-	-		-	-	-	-	1	3
CO5.**Interpret** and evaluate statistical and predictive models. (HOTS: Level 5: Evaluate).	2	3	-	2	3	-	-	-	-	-	-	-	-	3
CO6. **Conclude** the findings of predictive modelling. (HOTS: Level 5: Evaluate).	2	3	-	2	3	-	-	-	-	-	-	-	-	3
Level of Attainments PC/AIML/3-T														

ENVIRONMENTAL SCIENCE

<table>
<tr><td><b>Course code</b></td><td colspan="5"><b>MC/2-T</b></td></tr>
<tr><td><b>Category</b></td><td colspan="5"><b>Mandatory Courses</b></td></tr>
<tr><td><b>Course title</b></td><td colspan="5"><b>Environmental Sciences</b></td></tr>
<tr><td rowspan="2"><b>Scheme and Credits</b></td><td><b>L</b></td><td><b>T</b></td><td><b>P</b></td><td><b>Credits</b></td><td rowspan="2"></td></tr>
<tr><td><b>3</b></td><td><b>0</b></td><td><b>0</b></td><td><b>0.0</b></td></tr>
<tr><td><b>Pre-requisites (if any)</b></td><td colspan="5"><b>-</b></td></tr>
<tr><td><b>Course Assessment Methods (Internal: 30; External: 70)</b></td><td colspan="5"><b>Internal examination:</b>
• <b>Two minor tests each of 20 marks</b>
• <b>Class Performance measured through percentage of lectures attended (4 marks)</b>
• <b>Assignments, quiz etc. (6 marks)</b>
<b>End semester examination:</b>
• <b>Nine questions are to be set by the examiner.</b>
• <b>Question number one will be compulsory and based on the entire syllabus. It will contain seven short answers type questions.</b>
• <b>Rest of the eight questions is to be set with a fair weightage of all the units.</b>
• <b>All questions will carry equal marks.</b>
• <b>The Students will be required to attempt 05 questions in all.</b></td></tr>
</table>

Course Outcomes

1. Students will be able to enhance and analyze human impacts on the environment.
2. Integrate concepts & methods from multiple discipline and apply to environmental problems.
3. Design and evaluate strategic terminologies and methods for subs table management of environmental systems.
4. Field studies would provide students first-hand knowledge on various local environment aspects which forms an irreplaceable tool in the entire learning process.

Course content

UNIT-I

Multidisciplinary nature of Environmental studies: Definition, scope and importance, need for public awareness; Concept, Structure and function of an ecosystem: Producers, consumers and decomposers, Energy flow in the ecosystem ,Ecological succession ,Food chains, Food webs and ecological pyramids; Introduction, types, characteristics features, structure and function of Forest ecosystem, Grassland ecosystem ,Desert ecosystem, Aquatic ecosystem (Ponds, Stream, lakes, rivers, oceans, estuaries); Biodiversity: Introduction, Definition: genetic, species and ecosystem diversity, Bio- geographical classification of India, Value of biodiversity: consumptive use, productive use, social ethical, aesthetic and option values; Biodiversity at global, national and local level, India as a mega- diversity nation, Hot-spot of biodiversity, Threats to biodiversity: habitat loss, poaching of wildlife, man-wildlife conflicts, Endangered and endemic species of India, Conservation of biodiversity: In-situ and Ex-situ conservation of biodiversity.

UNIT-II

Renewable and non-renewable resources, Natural resources and associated problems ,Forest resources: Use and over-exploitation, deforestation, case studies, Timber extraction, mining, dams and their effects on forests and tribal people; Water resources: Use and over utilization of surface and ground water, floods, droughts conflicts over water, dams benefits and problems; Mineral resources: Use and exploitation, environmental effects of extracting and mineral resources; Food resources: World food problem, changes caused by agriculture and overgrazing, effects of modern agriculture, fertilizer- pesticide problems, water logging, salinity; Energy resources: Growing energy needs, renewable and non-renewable energy sources, use of alternate energy sources, case studies; Land resources: Land as a resource, land degradation, main induced landslides, soil erosion and desertification, Role of an individual in conservation of natural resources, Equitable use of resources for suitable lifestyle.

UNIT-III

Definition of Environment Pollution; Causes, effects and control measures of: Air Pollution, Water Pollution, Soil pollution, Marine pollution, Noise pollution, Thermal pollution, Nuclear hazards; Solid waste Management: Causes effects and control measures of urban and industrial wastes; Role of and individual in prevention of pollution, Pollution case studies; Disaster management: floods, earthquake, cyclone and landslides; Climate change, global warming, acid rain, ozone layer depletion, nuclear accidents and holocaust, Case studies; different laws related to environment: Environment Protection Act, Air (Prevention and Control of Pollution) Act, Water (Prevention and Control of Pollution) Act, Wildlife Protection Act, Forest Conservation Act.; Issues involved in enforcement of environmental legislation, Public awareness

UNIT-IV

Social issues and the Environment: From unsustainable to Sustainable development, Urban problems related to energy; Water conservation, rain water harvesting, watershed management; Resettlement and rehabilitation of people; its problem and concern, case studies; Environment ethics: Issues and possible solutions; Wasteland reclamation; Consumerism and waste products; Human Population growth, variation among nation, Population explosion-Family Welfare Programme, Environment and human health , Human Rights, Value Education, HIV/AIDS, Women and Child Welfare, Role of Information Technology in Environment and human health, Case Studies.

Books:

1. Fundamental concepts in Environmental studies by Dr. D.D. Mishra. S. Chand publications.
2. Essentials of Ecology and Environmental Science by Dr. S .V .S. Rana, PHI Learning Pvt. Ltd, Delhi
3. Environmental Chemistry by Anil Kumar De, Wiley Eastern Limited.
4. Environmental Science by T.G. Miller, Wadsworth Publishing Co, 13th edition.

Ecology and Environment by P. D. Sharma, Rastogi publication

FUNDAMENTALS OF AI & ML (PC/AIML/4-T)

General Course Information

Course Code: PC/AIML/4-T Course Credits: 3 Type: Professional Core Contact Hours: 3hours/week Mode: Lectures (L) Examination Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Basic Knowledge of Algorithms and Probability.

About the Course:

Artificial Intelligence is a core and an essential course for every graduate in Computer Science and Engineering. This course introduces the concepts of Artificial Intelligence and challenges inherent in building intelligent systems. It includes the role of knowledge representation in problem solving and how these are used in making intelligent machine. Further it incorporates the concepts of expert system and its applications.

Course Outcomes: By the end of the course students will be able to:

CO1. **Outline** various Artificial Intelligence techniques. (LOTS: Level 1: Remember)

CO2. **Illustrate** reasoning under uncertainty. (LOTS: Level 2: Understand)

CO3. **apply** search and knowledge representation techniques to solve AI problems.(LOTS: Level 3: Apply)

CO4. **Compare** strengths and weaknesses of AI algorithms (HOTS: Level 4: Analyse).

CO5. **Combine** various AI techniques to solve intelligent systems' problems. (HOTS: Level 6: Create)

Course Content

Unit – I

Introduction to AI: What is AI, Turing test, cognitive modeling approach, law of thoughts, the relational agent approach, the underlying assumptions about intelligence, techniques required to solve AI problems, level of details required to model human intelligence, successfully building an intelligent problem, history of AI

Unit II

Introduction to Machine Learning: What is Machine Learning, Learning from Data, History of Machine Learning, Big Data for Machine Learning, Leveraging Machine Learning, Descriptive vs Predictive Analytics, Machine Learning and Statistics, Artificial Intelligence and Machine Learning, Types of Machine Learning – Supervised, Unsupervised, Semi-supervised, Reinforcement Learning, Types of Machine Learning Algorithms, Classification vs Regression Problem, Bayesian, Clustering, Decision Tree, Dimensionality Reduction, Neural Network and Deep Learning, Training machine learning systems

Unit III

AI Research Trends: Research trends in machine learning, deep learning, reinforcement learning, robotics, computer vision, natural language processing, collaborative systems, algorithmic game theory, internet of things (Io T), neuromorphic computing
Applications of AI by domain: Transportation, home/service robots, healthcare, education, low- resource communities, public safety and security, employment and workplace, entertainment, finance, banking and insurance

Unit IV

Role of Artificial Intelligence in Society: Societal challenges AI presents, Ethical and Societal implications, policy and law for AI, fostering dialogue, sharing of best practices

Malicious Use of AI: Prevention and Mitigation: Security relevant properties of AI, Security domains and scenarios: digital security, physical security, political security, factors affecting the equilibrium of AI and security
Explainable AI: Introduction to explainable AI, why explainable AI, interpretability and explain ability, methods of interpretability and explain ability

Introduction to Data Analytics: Working with Formula and Functions, Introduction to Charts, Logical functions using Excel, Analyzing Data with Excel.

Text and Reference Books:

1. Artificial Intelligence 3e: A Modern Approach Paperback – By Stuart J Russell & Peter Norvig; Publisher – Pearson
2.Artificial Intelligence Third Edition By Kevin Knight, Elaine Rich, B. Nair – Mc Graw Hill
3Artificial Intelligence Third Edition By Patrick Henry Winston – Addison-Wesley Publishing Company
4 Machine Learning using Pytho, U Dinesh Kumar, Manaranjan Pradhan, John Wiley & Sons.
5 A Classical Approach to Artificial Intelligence, M. C. Trivedi, Khanna Publishing House.
6 Machine Learning, V. K. Jain, Khanna Publishing House.
7 Advanced Data Analytics Using Python: With Machine Learning, Deep Learning, Sayan Mukhopadhyay, Apress.
8 Machine Learning for Absolute Beginners: A Plain English Introduction, 2nd ed., Oliver Theobal
9 Big Data and Analytics, S. Acharya, S. Chellappan, Wiley Publication.
10. Introduction to Machine Learning, Jeeva Jose, Khanna Publishing House.

CO-PO Articulation Matrix: FUNDAMENTALS OF AIML(PC/AIML/4-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1. **Outline** various Artificial Intelligence techniques. (LOTS: Level 1: Remember)	1	–	–	–	–	–	–	–	–	–	–	–	–	3
CO2. **Illustrate** reasoning under uncertainty. (LOTS: Level 2: Understand)	1	–	–	–	–	–	–	–	–	–	–	–	–	3
CO3. **Apply** search and knowledge representation techniques to solve AI problems. (LOTS: Level 3: Apply)	2	2	–	2	2	–	–	–	–	–	–	–	–	3
CO4. **Compare** strengths and weaknesses of AI algorithms (HOTS: Level 4: Analyse).	2	2	2	2	–	–	–	–	–	–	–	–	–	3
CO5. **Combine** various AI techniques to solve intelligent systems' problems. (HOTS: Level 6: Create)	3	3	3	3	2	2	–	–	–	–	–	3	–	3
Level of Attainments PC/AIML/4-T														

DATA STRUCTURES AND ALGORITHMS USING C/C++LAB.

General Course Information

Course Code: PC/AIML/1-P Course Credits: 2 Type: Professional Core Lab. Course Contact Hours: 4 hours/week Mode: Lab practice and assignments	**Course Assessment Methods (internal: 50; external: 50)** The internal and external assessment is based on the level of participation in lab. Sessions and the timely submission of lab experiments/assignments, the quality of solutions designed for the assignments, the performance in VIVA-VOCE, the quality of lab. file and ethical practices followed. The internal examination is conducted by the course coordinator. The external examination is conducted by external examiner appointed by the Controller of Examination in association with the internal examiner appointed by the Chairperson of the Department.

Pre-requisites: Programming in C language.

About the Course:

This lab. Course involves implementation of basic and advance data structures and various operations on these data structures. The objective of the lab course is to train the students to solve the problems related to data structures and choose the appropriate data structure for solving computational problem efficiently.

Course Outcomes: By the end of the lab course a student would be able to:

CO1. **Implement** various data structures and the related operations. (LOTS: Levels 3: Apply)

CO2. **Analyse** space and time complexity of algorithms. (HOTS: Level 4: Analyse)

CO3. **Compare** solutions on the basis of the appropriateness of data structure used and the efficiency of the operations implemented. (HOTS: Level 5: Evaluate)

CO4. **Integrate** knowledge of data structures to solve real world problems related to data structure and algorithms. (HOTS: Level 6: Create)

CO5. **Create** written records for the given assignments with problem definition, design of solution and conclusions. (HOTS: Level 6: Create)

CO6. **Demonstrate** ethical practices while solving problems individually or in groups (LOTS: Level 3: Apply).

List of experiments/assignments

1. Two assignments related to creating and manipulating matrices and linear lists.
2. Two assignments associated with linked list, operations on linked lists and their applications.
3. Two assignments on array and linked implementation of stacks and queues.
4. Two assignments on trees and their applications.
5. Two assignments on graphs and their applications.
6. Two assignments on different searching and sorting methods along with their complexity analysis.
7. One assignment on challenging problems on data structures to be given in groups.

Note:

The actual experiments/assignments will be designed by the course coordinator. One assignment should be designed to be done in groups of two or three students. The assignments must meet the objective of the course and the levels of the given course outcomes. The list of assignments and schedule of submission will be prepared by the course coordinator at the beginning of the semester.

CO-PO Articulation Matrix: Data Structures and Algorithms using C/C++Lab. (PC/AIML/1-P)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1. **Implement** various data structures and the related operations. (LOTS: Levels 3: Apply)	2	-	-	-	2	-	-	-	2	-	-	-	3	-
CO2. **Analyse** space and time complexity of algorithms. (HOTS: Level 4: Analyse)	2	2	-	-	2	-	-	-	1	-	-	-	3	-
CO3. **Compare** solutions on the basis of the appropriateness of data structure used and the efficiency of the operations implemented. (HOTS: Level 5: Evaluate)	2	2	-	-	3	-	-	-	1	-	-	-	3	-
CO4. **Integrate** knowledge of data structures to solve real world problems related to data structure and algorithms. (HOTS: Level 6: Create)	3	2	3	-	3	-	-	-	3	-	-	-	3	-
CO5. **Create** written records for the given assignments with problem definition, design of solution and conclusions. (HOTS: Level 6: Create)	-	-	-	-	-	-	-	-	-	3	-	-	-	-
CO6. **Demonstrate** ethical practices while solving problems individually or in groups (LOTS: Level 3: Apply).	-	-	-	-	-	-	-	3	-	-	-	3	-	-
Level of Attainments: PC/AIML/1-P														

OBJECT ORIENTED PROGRAMMING USING C++ LAB.

General Course Information

Course Code: PC/AIML/2-P Course Credits: 2 Type: Professional Core Lab. Course Contact Hours: 4 hours/week Mode: Lab practice and assignments	**Course Assessment Methods (internal: 50; external: 50)** The internal and external assessment is based on the level of participation in lab. Sessions and the timely submission of lab experiments/assignments, the quality of solutions designed for the assignments, the performance in VIVA-VOCE, the quality of lab. file and ethical practices followed. The internal examination is conducted by the course coordinator. The external examination is conducted by external examiner appointed by the Controller of Examination in association with the internal examiner appointed by the Chairperson of the Department.

Pre-requisites: Problem solving using C Lab.

About the course:

The lab course provides the opportunity to students to solve problems using Object Oriented Framework in C++ language. This includes implementing the concepts of data abstraction, data hiding, and encapsulation, reuse of code and, compile and runtime polymorphism.

Course Outcomes: By the end of the course students will be able to:

CO1. **Implement** problems with object oriented framework. (LOTS: Level 3: Apply)

CO2. **Analyse** the structure of programs for modular design. (HOTS: Level 4: Analyse)

CO3. **Evaluate** robustness of a program by testing it on test/use cases. (HOTS: Level 5: Evaluate)

CO4. **Design** class hierarchies for implementing inheritance/polymorphism. (HOTS: Level 6: Create)

CO5. **Create** a lab record of assignments including problem definitions, design of solutions and conclusions. (HOTS: Level 6: Create)

CO6. **Demonstrate** ethical practices and solve problems individually or in a group. (LOTS: Level 3: Apply)

List of experiments

1. Create two classes **DM** and **DB** which store the value of distances. **DM** stores distances in meters and centimeters and **DB** in feet and inches. Write a program that can read values for the class objects and add one object of **DM** with another object of **DB**. Use a friend function to carry out the addition operation. The object that stores the results maybe a **DM** object or **DB** objects, depending on the units in which the result is required. The display should be in the format of feet and inches or meters and centimeters depending on the object on display.
2. Create a class rational which represents a numerical value by two double values-NUMERATOR & DENOMINATOR. Include the following public member Functions:
   - Constructor with no arguments (default).
   - Constructor with two arguments.
   - void reduce () that reduces the rational number by eliminating the highest common factor between
   - The numerator and denominator.
   - Overload + operator to add two rational number.
   - Overload >> operator to enable input through cin.
   - Overload << operator to enable output through cout.

   Write the main () function to test all the functions in the class.
3. A hospital wants to create a database regarding its indoor patients. The information to be stored includes
   a) Name of the patient b) Date of admission c) Disease d) Date of discharge
   Create a structure to store the date (year, month and day as its members). Create a base class to store the above information. The member function should include functions to enter information and display a list of all the patients in the database. Create a derived class to store the age of the patients. List the information about all the pediatric patients (less than twelve years in age).

4. Make a class **Employee** with a name and salary. Make a class **Manager** inherited from **Employee**. Add an instance variable named department of type string. Supply a method to **toString** that prints the manager's name, department and salary. Make a class **Executive** inherited from **Manager**. Supply a method **toString** that prints the string **"Executive"** followed by the information stored in the **Manager** superclass object. Supply a test program that tests these classes and methods.

5. Imagine a tollbooth with a class called 'tollBooth'. The two data items are of type unsigned int to hold the total number of cars, and a type double to hold the total amount of money collected. A constructor initializes both to 0. A member function called 'payingCar()' increments the car total and adds 0.50 to the cash total. Another function, called 'nopayCar ()', increments the car total but adds nothing to the cash total. Finally, a member function called displays the two totals. Include a program to test this class. This program should allow the user to push one key to count a paying car, and another to count a nonpaying car. Pushing the ESC key should cause the program to print out the total cars and total cash and then exit.

6. Write a function called 'revers_it()' that reverses a string (an array of char). Use a for loop that swaps the first and last characters, then the second and next to last characters and so on. The string should be passed to 'revers_it ()' as an argument. Write a program to exercise 'revers_it ()'. The program should get a string from the usercallof'revers_it () function and print out the result. Use an input method that allows embedded blanks. Test

the program with phrase, "*Guru Jambheshwar University of Science & Technology, Hisar*".

7. Write a program related to file handling with all the exception handling provisions.
8. C++ program to write and read time in/from binary file using fstream. Use exception handling wherever possible.
9. Write a program to implement string class using STL.
10. Write a program to implement run time polymorphism.

Note:

The experiments/assignments may vary from session to session and will be designed by the course coordinator. The assignments must meet the objective of the course and the levels of the given course outcomes. The course coordinator will provide the schedule for submission of the assignment.

CO-PO Articulation Matrix:Object Oriented Programming using C++ Lab. (PC/AIML/2-P)

List of Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
CO1. **Implement** problems with object oriented framework. (LOTS: Level 3: Apply)	2	2	-	-	1	-	-		2	-	-	2	3	-
CO2. **Analyse** the structure of programs for modular design. (HOTS: Level 4: Analyse)	2	2	-	-	2	-	-	-	-	-	-	-	3	-
CO3. **Evaluate** robustness of a program by testing it on test/use cases. (HOTS: Level 5: Evaluate)	2	2	-	-	2	-	-	-	-	-	-	-	3	-
CO4. **Design** class hierarchies for implementing inheritance/polymorphism. (HOTS: Level 6: Create)	3	-	1	-	2	-	-	-	-	-	-	2	3	-
CO5. **Create** a lab record of assignments including problem definitions, design of solutions and conclusions. (HOTS: Level 6: Create)	-	-	-	-	-	-	-	-	-	3	-	-	-	-
CO6. **Demonstrate** ethical practices and solve problems individually or in a group. (LOTS: Level 3: Apply)	-	-	-	-	-	-	-	3	3	-	-	3	-	-
Level of Attainments: PC/AIML/2-P														

DATA ANALYTICS USING R LAB.

General Course Information

Course Code: PC/AIML/3-P Course Credits: 2 Type: Professional Core Lab. Course Contact Hours: 4 hours/week Mode: Lab practice and assignments	**Course Assessment Methods (internal: 50; external: 50)** The internal and external assessment is based on the level of participation in lab. Sessions and the timely submission of lab experiments/assignments, the quality of solutions designed for the assignments, the performance in VIVA-VOCE, the quality of lab. file and ethical practices followed. The internal examination is conducted by the course coordinator. The external examination is conducted by external examiner appointed by the Controller of Examination in association with the internal examiner appointed by the Chairperson of the Department.

Pre-requisites: Basic programming skills.

About the Course:

The objective of this lab is to enable students to apply advanced data analytics tools for manipulating data, applying statistics, regression and classification.

Course Outcomes: By the end of the course students will be able to:

CO1. **Apply** pre-processing techniques to real world data. (LOTS: Level 3: Apply)

CO2. **Solve** problems of predictive analytics. (LOTS: Level 3: Apply)

CO3. **Evaluate** the performance of predictive models. (LOTS: Level 5: Evaluate)

CO4. **Design** complete data analytics experiments. (LOTS: Level 6: Create)

CO5. **Create** lab assignment record that includes problem definitions, solutions, results and conclusions. (HOTS: Level 6: Create).

CO6. **Demonstrate** ethical practices, self-learning and team spirit.

List of experiments/assignments using R:

1. Four Assignments on descriptive statistics
2. Four Assignment on visualizing data
3. Four Assignments on Pre-processing Data
4. Two assignments to solve linear and non-linear regression problems.
5. Two assignments on classification problems.
6. Two assignments on different sampling techniques.

Note:

The actual experiments/assignments will be designed by the course coordinator. One assignment should be designed to be done in groups of two or three students. The assignments must meet the objective of the course and the levels of the given course outcomes. The list of assignments and schedule of submission will be prepared by the course coordinator at the beginning of the semester.

CO-PO Articulation Matrix:Data Analytics using R Lab. (PC/AIML/3-P)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1. **Apply** pre-processing techniques to real world data. (LOTS: Level 3: Apply)	2	1	_	_	2	_	_	_	_	_	_	_	_	3
CO2. **Solve** problemsof predictive analytics. (LOTS: Level 3: Apply)	3	2	_	_	3	_	_	_	_	_	_	_	_	3
CO3. **Evaluate** the performance ofpredictive models. (LOTS: Level 5: Evaluate)	3	2	1	_	3	_	_	_	_	_	_	_	_	3
CO4. **Design**completedata analytics experiments. (HOTS: Level 6: Create)	3	2	2	3	3	_	_	_	_	_	_	_	_	3
CO5. **Create** lab assignment record that includes problem definitions, solutions, results and conclusions. (HOTS: Level 6: Create)	_	_	_	_	_	_	_	_	_	3	_	_	_	3
CO6. **Demonstrate** ethical practices, self-learning and team spirit. (LOTS: Level 3: Apply)	_	_	_	_	_	_	_	3	3	_	_	3	_	_
Level of Attainments PC/AIML/3-P														

COMPUTER ORGANISATION AND ARCHITECTURE

General Course Information

Course Code: PC/AIML/5-T Course Credits: 3 Type: Professional Core Contact Hours: 3 hours/week Mode: Lectures Examination Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Fundamental of Computer Systems.

About the Course:

Computer Architecture and organization describes the role of instruction set architecture in digital computer, main memory, and input/output devices. It illustrates the simple data path and control design for processors. It helps to understand the different operations and concept of instructions. It would enable the students to learn the basic function and architecture of modern computer systems.

Course Outcomes: By the end of the course students will be able to:

CO1. **Outline** the general concepts of digital electronics and computer organisation and architecture. (LOTS: Level 1: Remember)

CO2. **discuss** the basic components and their interfacing.(LOTS: Level 2: Understand)

CO3. **Apply** instructions for performing different operations. (LOTS: Level 3: Apply)

CO4. **analyse** the effect of addressing modes on the execution time of a program.(HOTS: Level 4: Analyse)

CO5. **Contrast** different types of memory, their architecture and access methods. (HOTS: Level 5: Evaluate)

CO6. **Design of** simple computer with different instruction sets. (HOTS: Level 6: Create)

Course Content

Unit I

Basic Principles: Boolean algebra and Logic gates, Combinational logic blocks (Adders, Subtractors, Multiplexers, Encoders, decoders, demultiplexers, KMaps), Sequential logic blocks (Flip-Flops, Registers, Counters); Flynn's classification of computers (SISD, MISD, MIMD); Performance metrics: MIPS, MFLOPS; CPU Architecture types: computer register, (accumulator, register, stack, memory/ register) detailed data path of a typical register based CPU.

Unit II

Computer Organization: Store program control concept, Instruction codes, timing and control, instruction cycle; type of instructions: memory reference, register reference, I/O reference; Basics of Logic Design, accumulator logic, Control memory; Micro Programmed Control: address sequencing, micro-instruction formats, micro-program sequencer, Implementation of control unit.

Unit III

Instruction Set Architecture & Parallelism: Instruction set based classification of processors (RISC, CISC, and their comparison); Stack Organization, Instruction Formats; addressing modes: register, immediate, direct, indirect, indexed; Operations in the instruction set: Arithmetic and Logical, Data Transfer, Control Flow; Types of interrupts; Introduction to Parallelism: Goals of parallelism (Exploitation of concurrency, throughput enhancement); Amdahl's law; Instruction level parallelism (pipelining, super scaling –basic features); Processor level parallelism (Multiprocessor systems overview).

Unit IV

Memory Hierarchy & I/O Techniques: The need for a memory hierarchy (Locality of reference principle, Memory hierarchy in practice: Cache, main memory and secondary memory, Memory parameters: access/ cycle time, cost per bit); Main memory (Semiconductor RAM & ROM organization, memory expansion, Static & dynamic memory types); Cache memory (Associative & direct mapped cache organizations; input-output interface, mode of transfer, DMA (Direct memory transfer).

Text and Reference Books:

1. Mano, M. Morris, *Digital Logic and Computer Design*, Prentice Hall of India Pvt. Ltd., 1981.
2. M. Morris Mano, *Computer System Architecture*, Prentice Hall of India Pvt. Ltd., 1993.
3. Milles J. Murdocca, Vincent P. Heuring, *Computer Architecture and Organization, An Integrated Approach*, JohnWiley& Sons Inc., 2007.

4. William Stallings, 10th edition, *Computer Organization and Architecture*, Prentice Hall, 2016.
5. Heuring, V.P., Jordan, H.F., *Computer Systems Design and Architecture*, Addison Wesley,1997.
6. R.P Jain, *Modern Digital Electronics*, 3rd Edition , Tata McGraw Hill,, 2003.

CO-PO Articulation Matrix:Computer Organisation and Architecture (PC/AIML/5-T)

List of Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
CO1. **outline** the general concepts of digital electronics and computer organisation and architecture. (LOTS: Level 1: Remember)	1	-	-	-	-	-	-	-	-	-	-	-	2	-
CO2. **discuss** the basic components and their interfacing. (LOTS: Level 2: Understand)	1	-	-	-	-	-	-	-	-	-	-	-	3	-
CO3. **Apply** instructions for performing different operations. (LOTS: Level 3: Apply)	2	-	-	-	-	-	-	-	-	-	-	-	3	-
CO4. **Analyse** the effect of addressing modes on the execution time of a program. (HOTS: Level 4: Analyse)	2	2	-	1	-	-	-	-	-	-	-	1	3	-
CO5. **Contrast** different types of memory, their architecture and access methods. (HOTS: Level 5: Evaluate)	2	2	-	1	-	-	-	-	-	-	-	1	3	-
CO6. **Design** of simple computer with different instruction sets. (HOTS: Level 6: Create)	3	2	-	-	2	-	-	-	-	-	-	-	3	-
Level of Attainments PC/AIML/5-T														

DATA MINING TECHNIQUES

General Course Information

Course Code: PC/AIML/6-T Course Credits: 3 Type: Contact Hours: Mode: Examination Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Knowledge of database systems, elementary knowledge of statistics and probability.

About the Course:

Today's era is the era of information. Data is growing exponentially day by day. There is a need to process and analyse the data to extract knowledge from it, so that one can use that knowledge for decision making. This course provides introductory concepts of data mining and data warehousing. The course will be taught with a database as well as machine learning perspectives. The objective of the course is to provide a comprehensive understanding of data mining tasks and evaluation of results obtained out of data mining processes.

Course Outcomes: By the end of the course students will be able to:

CO1. **Outline** various types of data mining and data warehouse concepts and techniques. (LOTS: Level 1: Remember)

CO2. **Explain** association of patterns, data mining functionalities, tasks of data mining, (LOTS: Level 2: Understand)

CO3. **Apply** various classifications, clustering correlation and association mining for extracting valuable information from data. (LOTS: Level 3: Apply)

CO4. **Evaluate** the descriptive and predictive data mining models. (HOTS: Level 5: Evaluate)

CO5. **Plan** a data mining process for discovering knowledge from real-world databases. (HOTS: Level 6: Create)

Course Content

Unit I

Introduction to Data Mining: Kind of data to be mined, Data Mining Functionalities, Technologies used in Data Mining, Applications of data Mining, Major Issues in Data Mining.
Data Warehouse: Introduction, Data Warehouse and Database Systems, Data Warehouse Architecture, Data Warehouse Models, Data Cube and OLAP, Multidimensional data Model, Concept Hierarchies, OLAP operations
Pattern Mining: Mining Frequent Patterns, Associations and Correlations, Frequent Item_set Mining using Apriori Algorithm, Generating Association Rules from Frequent Itemsets. Pattern Growth Approach for Mining Frequent Itemsets, Pattern evaluation Methods

Unit II

Classification: Introduction, Classification using Decision Tree Induction, Bayesian Classification Methods, Rule Based Classification, Model Evaluation and Selection, Techniques to Improve Classification Accuracy.
Introduction to advanced classifiers: k-Nearest Neighbor, Support Vector Machine, Artificial Neural Network.

Unit III

Cluster Analysis: Introduction, overview of Basic Clustering Methods,
Partitioning Methods: k-mean, k-medoids,
Hierarchical Methods: Agglomerative versus Divisive Hierarchical Clustering, Distance Measures in Algorithmic Methods, Balanced Iterative Reducing and Clustering using Hierarchies (BIRCH), Chameleon: Multiphase Hierarchical Clustering Using Dynamic Modelling, Probabilistic Hierarchical Clustering,
Density-based methods: DBSCAN, OPTICS, DENCLUE,
Grid-based Methods: STING, CLIQUE, **Evaluation of Clustering.**

Unit IV

Outlier Detection: Introduction, types of outliers, challenges of outlier detection.
Outlier detection methods: statistical approaches, proximity-based approaches, clustering based approaches, classification-based approaches, Outlier detection in high dimensional data.

Text and Reference Books:

1. Jiawei Han, Micheline Kamber and Jian Pei, *Data Mining Concepts and Techniques,* Morgan Kaufmann Publishers,Third Edition, July 2011.
2. Alex Berson, Stephen J. Smith, *Data Warehousing, Data Mining & OLAP*, Tata McGraw Hill, 2004.
3. Pang-Ning Tan, Michael Steinbach and Vipin Kumar, *Introduction to Data Mining*, Pearson Education, 2014.

4. K. P. Soman, Shyam Diwakar and V. Ajay, *Insight into Data Mining Theory and Practice*, Easter Economy Edition, Prentice Hall of India, 2009.
5. G. K. Gupta, *Introduction to Data Mining with Case Studies*, Prentice Hall of India, 2006.
6. Daniel T. Larose, *Data Mining Methods and Models*, Wiley, 2006.
7. W. H. Inman, *Building the Data Warehouse*, Wiley India, 2005

CO-PO Articulation Matrix:Data Mining Techniques (PC/AIML/6-T)

List of Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
CO1. **Outline** various types of data mining and data warehouse concepts and techniques. (LOTS: Level 1: Remember)	1	_	_	_	2	_	_	_	_	_	_	_	_	1
CO2. **Explain** characteristics, architecture of a data warehouse, OLAP operations and data mining tasks. (LOTS: Level 2: Understand)	1	_	_	_	3	_	_	_	_	_	_	_	_	2
CO3. **Apply** various pre-processing and data mining techniques for extracting valuable information from data. (LOTS: Level 3: Apply)	2	1	_	_	3	_	_	_	_	_	_	_	_	3
CO4. **Evaluate** the descriptive and predictive data mining models. (HOTS: Level 5: Evaluate)	3	2	2	3	3	_	_	_	_	_	_	_	_	3
CO5. **Plan** a data mining process for discovering knowledge from real-world databases. (HOTS: Level 6: Create)	3	3	3	3	3	_	_	_	_	_	_	_	_	3
Level of Attainments PC/AIML/6-T														

OPERATING SYSTEMS

General Course Information

Course Code: PC/AIML/7-T Course Credits: 3 Type: Professional Core Contact Hours: 3hours/week Mode: Lectures (L) Examination Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Programming in C and knowledge of computer fundamentals.

About the Course:

The objective of this course is to help students become familiar with the fundamental concepts of operating systems and provide them with enough understanding of operating system design.

Course Outcomes: By the end of the course students will be able to:

CO1. **list** various functions and design characteristics of operating systems (LOTS: Level 1: Remember)

CO2. **Explain** fundamental concepts of operating systems. (LOTS: Level 2: Understand)

CO3. **apply** operating system design concepts for solving problems regarding scheduling, memory management, disk management and deadlocks etc.(LOTS: Level 3: Apply)

CO4. **Analyze** the issues related to various operating systems. (HOTS: Level 4: Analyze)

CO5. **Design** solutions for the memory and process management problems. (HOTS: Level 6: Create)

Course Content

Unit I

Introductory Concepts: Operating systems functions and characteristics, operating system services and systems calls, system programs, operating system structure. operating systems generation, operating system services and systems calls. Types of Operating systems: Batch operating system, Time-sharing OS, Distributed operating system, Realtime systems.

File Systems: Types of Files and their access methods, File allocation methods, Directory Systems: Structured Organizations, directory and file protection mechanisms, disk scheduling and its associated algorithms.

Unit II

Processes: Process concept, Process Control Block, Operations on processes, cooperating processes.CPU scheduling: Levels of Scheduling, scheduling criteria, Comparative study of scheduling algorithms, Algorithm evaluation, multiple processor scheduling. Critical-section problem, Semaphores.

Unit III

Storage Management: Storage allocation methods: Single contiguous allocation, non-contiguous memory allocation, Paging and Segmentation techniques, segmentation with paging, Virtual memory concepts, Demand Paging, Page replacement Algorithms, Thrashing.

Unit IV

Deadlock: System model, Deadlock characterization, Methods for handling deadlocks, Deadlock prevention, Deadlock avoidance, Deadlock detection, Recovery from deadlock

Case Studies: Comparative study of WINDOW, UNIX & LINUX system.

Text and Reference Books:

1. Silberschatz, Peter B. Galvin and Greg Gagne, *Operating System Concepts*, 8th Edition, WileyIndian Edition, 2010.
2. Andrew S Tanenbaum, *Modern Operating Systems*, Third Edition,Prentice Hall India, 2008.
3. Naresh Chauhan, *Principles of Operating Systems*, Oxford Press, 2014.
4. D.M. Dhamdhere, *Operating Systems*, 2nd edition, Tata McGraw Hill, 2010.
5. William Stallings, *Operating Systems– Internals and Design Principles*, 5th Edition,Prentice Hall India, 2000.

CO-PO Articulation Matrix:Operating Systems (PC/AIML/7-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1. **List** various functions and design characteristics of operating systems (LOTS: Level 1: Remember)	1	_	_	_	_	_	_	_	_	_	_	_	3	_
CO2. **Explain** fundamental concepts of operating systems. (LOTS: Level 2: Understand)	1	_	_	_	_	_	_	_	_	_	_	_	3	_
CO3. **Apply** operating system design concepts for solving problems regarding scheduling, memory management, disk management and deadlocks etc. (LOTS: Level 3: Apply)	3	1	_	_	2	_	_	_	_	_	_	1	3	_
CO4. **Analyze** the issues related to various operating systems. (HOTS: Level 4: Analyse)	3	2	3		2	_	_	_	_	_	_	1	3	_
CO5. **Design** solutions for the memory and process management problems. (HOTS: Level 6: Create)	3	2	3	2	2	_	_	_	_	_	_	_	3	_
Level of Attainments PC/AIML/7-T														

DATABASE MANAGEMENT SYSTEM

General Course Information

Course Code: PC/AIML/8-T Course Credits: 3 Type: Professional Core Contact Hours: 3 hours/week Mode: Lectures (L) Exam Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Prerequisite: Knowledge of UNIX/ Windows, programming language and data structures

About the Course:

This course includes a detailed coverage of principles of database design and models. Students learn querying a database using SQL, normalization techniques, transaction processing etc.

Course Outcomes: By the end of the course students will be able to:

CO1. **Describe** fundamental elements of Database Management System. (LOTS: Level 1: Remember)

CO2. **Discuss** principles of relational Database modelling. (LOTS: Level 2: Understanding)

CO3. **Apply** SQL for designing queries for Relational Databases. (LOTS: Level 3: Apply)

CO4. **Contrast** various concurrency control and recovery techniques with concurrent transactions in DBMS. (HOTS: Level 5: Evaluate)

CO5. **design** models of databases using ER modelling and normalization for real life applications.(HOTS: Level 6: Create)

Course Content

Unit I

Overview: Overview of File Systems and Database Systems, Characteristics of the Data Base Approach, Database users, Advantages and Disadvantages of a DBMS, Responsibility of Database Administrator.

Data Base Systems Concepts and Architecture: DBMS architecture and various views of Data, Data Independence, Database languages, Data Models: Relational Database Model, Hierarchical Data Model, Network Data Model, Schemas and Instances.

Unit II

E-R Model: Entity Types, Attributes & Keys, Relationships, Roles and Structural Constraints, E-R Diagrams, Reduction of an E-R Diagram to Tables. Relational Model and Query Language: Overview of Relational Database, Key Integrity Constraints, Relational Algebra, Relational Calculus, SQL fundamentals, Basic Operators, Missing information and NULL values, Advanced SQL features

Unit III

Relational Database Design: Overview of normalization, Database Anomalies, Candidate and Super Key, Functional Dependencies, Integrity Constraints, Decomposition, Normal forms: First, Second, Third Normal, Boyce Codd, Normal Form, Multi-valued Functional Dependencies and Fourth Normal Form, Join Dependencies and Fifth Normal Form, Denormalization.

Unit IV

Concurrency Control Techniques: Overview of database Transactions, Transaction states, ACID properties of a Transaction, Transaction Recovery, Concurrency Control, Locking Techniques, Time-stamp ordering, Multi-version Techniques, Deadlock, Recovery Techniques in centralized DBMS.

DDBMS Design: Replication and Fragmentation Techniques.

Text and Reference Books:

1. Elmasri, R., and Navathe,S. B., *Fundamentals of Database Systems,* 3rd Edition, Addison Wesley, 2002.
2. Silberschatz, A., Korth, H. F., and Sudarshan, S., Database *System Concepts,* McGraw Hill, 2011.
3. Pannerselvam R., *Database Management Systems*, 2ndEdition, PHI Learning, 2011.
4. Desai, B. C., *An Introduction to Database System*,Galgotia Publication, 2010.
5. Leon, A., and Leon, M.,*Database Management Systems*,Ist Edition,Vikas Publishing, 2009.

6.Mata-Toledo, R., Cushman, P., Sahoo, D., *Database Management Systems*, Schaums'Outline series, TMH, 2007.

CO-PO Articulation Matrix:Database Management System (PC/AIML/8-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1. **Describe** fundamental elements of Database Management System. (LOTS: Level 1: Remember)	1	--	--	--	--	--	--	--	--	--	--	--	3	--
CO2. **discuss** principles of relational Database modeling. (LOTS: Level 2: Understanding)	1	--	--	--	--	--	--	--	--	--	--	--	3	--
CO3. **Apply** SQL for designing queries for Relational Databases. (LOTS: Level 3: Apply)	1	--	--	_	2	--	--	--	--	--	--	--	3	--
CO4. **Contrast** various concurrency control and recovery techniques with concurrent transactions in DBMS. (HOTS: Level 5: Evaluate)	1	2	--	--	--	--	--	--	--	--	--	--	3	--
CO5. **Design** models of databases using ER modelling and normalization for real life applications.(HOTS: Level 6: Create)	3	2	3	1	2	--	--	--	--	--	--	--	3	--
Level of Attainments PC/AIML/8-T														

DISCRETE MATHEMATICS

General Course Information

Course Code: PC/AIML/9-T Course Credits: 3 Type: Professional Core Contact Hours: 3hours/week Mode: Lectures (L) Examination Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Basic knowledge of Number Theory, Calculus and Algebra

About the Course:

Discrete Mathematics is a core and an essential course for every graduate in Computer Science and Engineering. This branch of mathematics mainly deals with discrete objects (as computer runs on discrete steps). It provides a mathematical language for computer science to resolve many real world problems by incorporating different methods applicable to various discrete structures. This course introduces set theory, propositional calculus, algebraic structures, recurrence relations and graph theory.

Course Outcomes: By the end of the course a student would be able to:

CO1. **Outline** various discrete structures and the related operations. (LOTS: Level 1: Remember)

CO2. **Illustrate** different discrete structures with the help of examples. (LOTS: Level 2: Understand)

CO3. **apply** appropriate techniques to solve problems related to discrete structures.(LOTS: Level 3: Apply)

CO4. **Justify** the solutions with the help of proofs. (HOTS: Level 5: Evaluate)

CO5. **Combine** techniques related to discrete structures for solving real world problems. (HOTS: Level 6: Create)

Course Content

Unit I

Set Theory: Introduction to Set Theory, Venn Diagrams, Set Operations, Algebra of Sets, Duality, Finite, Infinite Sets and Counting Principle, Classes of Sets, Power Sets, Partitions, Multi Sets, Relations: Cartesian Product, Representation of Relations, Types of Relation, Equivalence Relations and Partitions, Partial Ordering Relations, Functions: Definition, Types of Functions, Composition of Functions, Inverse Function, Recursively Defined Functions.

Unit II

Logic and Propositional Calculus: Introduction, Propositions and Compound Propositions, Basic Logical Operations, Propositions and Truth Tables, Tautologies and Contradictions, Logical Equivalence, Algebra of Propositions, Conditional and Bi-conditional Statements, Algebraic Structures: Group Axioms, Monoid, Semi-Groups, Subgroups, Abelian Group, Cosets, Normal Subgroup, Cyclic Group, Permutation Group, Lagrange’s Theorem, Homomorphism, Isomorphism, Automorphism, Rings, Integral Domains and Fields (Also, some basic and standard results related to Groups, Rings, ID and Fields).

Unit III

Recursion and Recurrence Relation: Polynomials and their evaluation, Sequences, Introduction to AP, GP and AG Series, Partial Fractions, Recurrence Relation, Linear Recurrence Relations with Constant Coefficients, Linear Homogeneous Recurrence Relations with Constant Coefficients, Particular Solution- Homogeneous Linear Difference Equations, Non-Homogeneous Linear Difference Equations, Total Solution, Generating Functions.

Unit IV

Graphs Theory: Introduction to Graphs, Multi Graph, Directed and Undirected Graphs, Subgraphs, Bipartite Graphs, Regular Graphs, Connected Graphs, Homomorphic and Isomorphic Graphs, Cut points and Bridges, Paths and Circuits, Euler Graph, Hamiltonian Graph, Planar Graph, Euler Formula, Weighted Graphs ,Dijkstra’s Shortest Path Algorithm for Weighted Graphs, Trees, Spanning Trees, Minimum Spanning Tree (Prim’s and Kruskal’s Algorithm).

Text and Reference Books:

1. J.P. Trembley and R. Manohar, *Discrete Mathematical Structures with Applications to Computer Science*, Tata McGraw Hill – 13th reprint, 2012.
2. Kenneth H. Rosen, *Discrete Mathematics and its applications*, 6th Edition, Tata McGraw Hill, 2011.
3. Richard Johnsonbaugh, *Discrete Mathematics*, 6th Edition, Pearson Education Asia, 2011.
4. S. Lipschutz and M. Lipson, *Discrete Mathematics*, Tata McGraw Hill, 3rd Edition, 2010.
5. B. Kolman, R. C. Busby and S. C. Ross, *Discrete Mathematical structures*, 6th Edition, PHI, 2010.
6. C. L. Liu, *Elements of Discrete Mathematics*, Tata McGraw Hill, 3rd Edition, 2008.

CO-PO Articulation Matrix:Discrete Mathematics (PC/AIML/9-T)

List of Course Outcomes	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1. **Outline** various discrete structures and the related operations. (LOTS: Level 1: Remember)	1	--	--	--	--	--	--	--	--	--	--	--	1	--
CO2. **Illustrate** different discrete structures with the help of examples. (LOTS: Level 2: Understand)	1	--	--	--	--	--	--	--	--	--	--	--	2	--
CO3. **Apply** appropriate techniques to solve problems related to discrete structures. (LOTS: Level 3: Apply)	2	--	--	--	1	--	--	--	--	--	--	--	3	2
CO4. **Justify** the solutions with the help of proofs. (HOTS: Level 5: Evaluate)	3	1	--	--	2	--	--	--	--	--	--	--	3	--
CO5. **Combine** techniques related to discrete structures for solving real world problems. (HOTS: Level 6: Create)	3	2	--	--	2	--	--	--	1	--	--	1	3	2
Level of Attainments: PC/AIML/9-T														

PYTHON PROGRAMMING (PC/AIML/10-T)

General Course Information

Course Code: PC/AIML/10-T/ Course Credits: 3 Type: Professional Core Contact Hours: 3hours/week Mode: Lectures (L) Examination Duration: 3 hours	**Course Assessment Methods (internal: 30; external: 70)** Two minor examinations (20 marks), Class Performance measured through percentage of lectures attended (4 marks), assignments (6 marks), and the end- semester examination (70 marks). For the end semester examination, nine questions are to be set by the examiner. A candidate is required to attempt 5 questions in all. All questions carry equal marks. Question number 1 will be compulsory and based on the entire syllabus. It will contain seven parts of 2 marks each. Question numbers 2 to 9 will be given by setting two questions from each of the four units of the syllabus. A candidate is required to attempt the remaining four questions by selecting one question from each of the four units.

Pre-requisites: Exposure to programming languages.

About the Course:

Python is a popular open source programming language used for both standalone programs and scripting applications in a wide variety of domains. It is free, portable, and powerful and is both relatively easy and remarkably fun to use. In today’s era Python has found great applicability in machine learning, data analytics and many other data science application. This is introductory course and covers most of the basic concepts required for basic python programming. Some of the contents are advanced may be useful for data analytics purpose.

Course Outcomes: By the end of the course students will be able to:

CO1. **Outline** various basic programming constructs including operators, character sets, basic data types and control statements. (LOTS: level 1: Understand)

CO2. **Explain** Python packages and their functionalities for data analysis. (LOTS: level 2: Understand)

CO3. **Solve** problems using python programming. (LOTS: level 3: Apply)

CO4. **Analyze** the results of data analysis or machine learning programs (HOTS: level 4: Analyze)

CO5. **Evaluate** solutions according to the problem definition. (HOTS: level 5: Evaluate)

CO6. **Develop** database applications in Python. (HOTS: level 6: Create)

Course Content

Unit I

Introduction to Python, History of Python, Features of Python, Python Identifiers, Python Character Set, Keywords and Indentation, Comments, Command Line Arguments, Assignment Operator, Operators and Expressions, *print()* Function, *input()* Function, *eval()* Function, Python Data Types: *int, float, complex*, Variables, Mutable vs Immutable variables, Namespaces, Decision Statements: Boolean Type, Boolean Operators, *if* statement, *else* statement, Nested Conditionals Statements, Multi-way Decision Statements (*elif* statement).

Unit II

Loop Control Statements: *While* loop, *range()* Function, *For* Loop, Nested Loops, Infinite Loop, *Break* Statement, *Continue* Statement, *Pass* Statement, Introduction to Strings, String Operations: Indexing and Slicing, Lists: Operations on List: Slicing, Inbuilt Functions for Lists, List Processing: Searching and Sorting, Dictionaries: Need of Dictionary, Operations on Directories: Creation, Addition, Retrieving Values, Deletion; Tuples, operations on Tuples, Inbuilt Functions for Tuples, Introduction to Sets, operations on sets.

Python Functions, Inbuilt functions, *Main* function, User Defined functions, Defining and Calling Function, Parameter Passing, Actual and Formal Parameters, Default Parameters, Global and Local Variables, Recursion, Passing Functions as Data, *Lamda* Function, Modules, Importing Own Module, Packages.

Unit III

Operations on File: Reading text files, read functions, *read()*, *readline()* and *readlines()*, writing Text Files, write functions, *write()* and *writelines()*,Manipulating file pointer using seek, Appending to Files.

Python Object Oriented: Overview of OOP, Classes and objects, Accessing attributes, Built-In Class Attributes, Methods, Class and Instance Variables, Destroying Objects, Polymorphism, Overlapping and Overloading of Operators, Class Inheritance: *super()*, Method Overriding, Exception Handling, *Try-except-else* clause, Python Standard Exceptions, User-Defined Exceptions

Unit IV

Databases in Python: Create Database Connection, *create, insert, read, update* and *delete* Operation, DML and DDL Operation with Databases.
Python for Data Analysis: *numpy*: Creating arrays, Using arrays and Scalars, Indexing Arrays, Array Transposition, Universal Array Function, Array Processing, Array Input and Output

Pandas: Series, Data Frame, Panel, Index objects, Re-indexing, Iteration, Sorting. *Matplotlib*: Python for Data Visualization, Visualization Section, *Sklearn*: loading of dataset, learning and predicting, Model Persistence.

Text and Reference Books:

1. Ashok Namdev Kamthane, *Programming and Problem Solving with Python*, Mc Graw Hill Education Publication, 2018.

2. John Guttag, *Introduction to Computation and Programming using Python*, Springer, Revised and Expanded version (Referred by MIT), 2013.

3. Lutz, M., *Learning Python: Powerful Object-Oriented Programming*. O'Reilly Media, Inc., 2013.

4. Michael T Goodrich and Robertto. Thamassia, Micheal S Goldwasser, *Data Structures and Algorithms in Python*, Wiley, 2016.

5. Y. Daniel Liang, *Introduction to Programming Using Python*, Pearson, 2013.

6. Reema Thareja, *Python Programming Using Problem Solving Approach* , Oxford Publications, 2017.

7. Dr. R. Nageswara Rao, Allen B. Downey, *Core Python Programming* , *Think Python*, O’Reilly Media, 2012.

8. Kenneth A. Lambert, *The Fundamentals of Python: First Programs*, Cengage Learning, 2011.

Python Programming Course (PC/AIML/10-T)

List of Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
CO1. **Outline** various basic programming constructs including operators, character sets, basic data types and control statements. (LOTS: level 1: Understand)	1	–	–	–	–	–	–	–	–	–	–	–	–	–
CO 2.Explain Python packages and their functionalities for data analysis. (LOTS: level 2: Understand)	1	–	–	–	–	3	–	–	–	–	–	–	–	–
CO 3. **Solve** problems using python programming. (LOTS: level 3: Apply)	3	2	-	2	-	–	–	–	–	–	–	–	–	-
CO 4. **Analyse** the results of data analysis or machine learning programs (HOTS: level 4: Analyse)	2	3	2	2	_	_	_	_	_	_	_	_	–	-
CO 5.**Evaluate** solutions according to the problem definition. (HOTS: level 5: Evaluate)	2	3	2	2	-	3	_	_	_	_	_	-	–	-
CO 6.**Develop** database applications in Python. (HOTS: level 6: Create	3	3	2	3	-	-	-	-	-	-	-	-	3	-
Level of Attainments PC/AIML/10-T														

DATA MINING USING R/PYTHON/WEKA LAB.(PC/AIML/6-P)

General Course Information

Course Code: PC/AIML/6-P Course Credits: 2 Type: Professional Core Lab. Course Contact Hours: 4 hours/week Mode: Lab practice and assignments	**Course Assessment Methods (internal: 50; external: 50)** The internal and external assessment is based on the level of participation in lab. Sessions and the timely submission of lab experiments/assignments, the quality of solutions designed for the assignments, the performance in VIVA-VOCE, the quality of lab. file and ethical practices followed. The internal examination is conducted by the course coordinator. The external examination is conducted by external examiner appointed by the Controller of Examination in association with the internal examiner appointed by the Chairperson of the Department..

Pre-requisites: Basic Programming skills.

About the Course:
The objective of this lab is to enable students to use tools for applying advanced data reduction, classification and clustering techniques.

Course Outcomes: By the end of the course students will be able to:

CO1. **Apply** advanced data mining algorithms. (LOTS: Level 3: Apply)

CO2. **Usages** of modern data mining tools such as WEKA, R/Python packages. (LOTS: Level 3: Apply)

CO3. **Evaluate** the performance of data mining models. (LOTS: Level 5: Evaluate)

CO4. **Design** advanced data mining experiments. (LOTS: Level 6: Create)

CO5. **Create** lab assignment record that includes problem definitions, solutions, results and conclusions. (HOTS: Level 6: Create).

CO6. **Demonstrate** ethical practices, self-learning and team spirit. (LOTS: Level 3: Apply)

List of experiments/assignments: (WEKA/R/Python packages etc.)

1. Five assignments on advanced classification algorithms (Advanced Classifiers).
2. Five assignment on clustering problems
3. Four assignment on data reduction and attribute selection
4. Two assignments on discovering association rules.

Note:
The actual experiments/assignments will be designed by the course coordinator. One assignment should be designed to be done in groups of two or three students. The assignments must meet the objective of the course and the levels of the given course outcomes. The list of assignments and schedule of submission will be prepared by the course coordinator at the beginning of the semester.

CO-PO Articulation Matrix:Data Mining using R/Python/WEKA Lab. (PC/AIML/6-P)

List of Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
CO1. **Apply** advanced data mining algorithms. (LOTS: Level 3: Apply).	2	3	3	–	3	–	–	–	–	–	–	–	–	3
CO2. **Usages** of modern data mining tools such as WEKA, R/Python packages. (LOTS: Level 3: Apply)	1	-	–	–	3	–	–	–	–	–	–	–	–	3
CO3. **Evaluate** the performance ofdata mining models (LOTS: Level 5: Evaluate)	3	2	–	–	3	–	–	–	–	–	–	–	–	3
CO4. **Design** advanced data mining experiments. (LOTS: Level 6: Create)	3	3	–	3	3	–	–	–	–	–	–	–	–	3
CO5. **Create** lab assignment record that includes problem definitions, solutions, results and conclusions. (HOTS: Level 6: Create)	–	–	–	–	–	–	–	–	–	3	–	–	–	3
CO6. **Demonstrate** ethical practices, self-learning and team spirit. (LOTS: Level 3: Apply)	–	–	–	–	–	–	–	3	3	–	–	3	–	–
Level of Attainments PC/AIML/6-P														

DATABASE MANAGEMENT SYSTEM LAB.(PC/AIML/8-P)

General Course Information

Course Code: PC/AIML/8-P Course Credits: 2 Type: Professional Core Lab. Course Contact Hours: 2 hours/week Mode: Lab practice and assignments.	**Course Assessment Methods (internal: 50; external: 50)** The internal and external assessment is based on the level of participation in lab. Sessions and the timely submission of lab experiments/assignments, the quality of solutions designed for the assignments, the performance in VIVA-VOCE, the quality of lab. file and ethical practices followed. The internal examination is conducted by the course coordinator. The external examination is conducted by external examiner appointed by the Controller of Examination in association with the internal examiner appointed by the Chairperson of the Department.

Pre-requisites: Exposure to programming language, MS Access.

About the Course:

This lab. course on DBMS involves a rigorous training on Oracle programming. It provides a strong formal foundation in database concepts, technology and practice to the students to groom them into well-informed database application developers. The objective of the lab course is to develop proficiency in the execution of commands of the database design and query using Oracle.

Course Outcomes: By the end of the course students will be able to:

CO1. **Implement** database problems using Oracle DML/DDL commands. (LOTS: Level 3: Apply)

CO2. **Enforce** integrity constraints on a database using a state-of-the-art RDBMS. (LOTS: Level 3: Apply)

CO3. **Analyse** the design of a relational database. (HOTS: Level 4: Analyse)

CO4. **Design** a relational database for a given schema. (HOTS: Level 6: Create)

CO5. **Create** lab assignment record that includes problem definitions, solutions, results and conclusions. (HOTS: Level 6: Create)

CO6. **Demonstrate** ethical practices, self-learning and team spirit.

List of experiments/assignments:

1. Use oracle software and login with valid user id and password. Explore its GUI and practice some basic commands of it.
2. Three assignments related to creation of database with tables having different fields and data types.
3. Two assignments on the creation of table with different types of constraints.
4. Two assignments on insert, delete and modify records from the tables.
5. Two assignments on modifying the table using the alter command.
6. Two assignments on exploring select statement using various clauses like where, order by, group by, having and aggregate functions.
7. Two assignments on the use of set operations to query the tables.
8. Two assignments on creating joins and views on the tables.
9. One assignment on generating sub-queries.

Note:

The actual experiments/assignments will be designed by the course coordinator. One assignment should be designed to be done in groups of two or three students. The assignments must meet the objective of the course and the levels of the given course outcomes. The list of assignments and schedule of submission will be prepared by the course coordinator at the beginning of the semester.

CO-PO Articulation: Matrix Database Management System Lab. (PC/AIML/8-P)

List of Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
CO1. **Implement** database problems using Oracle DML/DDL commands. (LOTS: Level 3: Apply)	2	1		_	2	_	_	_	_	_	_	_	3	_
CO2. **enforce** integrity constraints on a database using a state-of-the-art RDBMS. (LOTS: Level 3: Apply)	2	2	_	_	2	_	_	_	_	_	_	_	3	_
CO3. **Analyse** the design of a relational database. (HOTS: Level 4: Analyse)	3	3	1	_	2	_	_	_	_	_	_	_	3	_
CO4. **Design** a relational database for a given schema. (HOTS: Level 6: Create)	3	3	2	3	3	_	_	_	_	_	_	_	3	_
CO5. **Create** lab assignment record that includes problem definitions, solutions, results and conclusions. (HOTS: Level 6: Create)	_	_	_	_	_	_	_	_	_	3	_	_	_	_
CO6. **Demonstrate** ethical practices, self-learning and team spirit. (LOTS: Level 3: Apply)	_	_	_	_	_	_	_	3	3	_	_	3	_	_
Level of Attainments PC/AIML/8-P														

PYTHON PROGRAMMING LAB.(PC/AIML/10-P)

General Course Information

Course Code: PC/AIML/10-P Course Credits: 1.5 Type: Professional Core Lab. Course Contact Hours: 3 hours/week Mode: Lab practice and assignments	**Course Assessment Methods (internal: 50; external: 50)** The internal and external assessment is based on the level of participation in lab. Sessions and the timely submission of lab experiments/assignments, the quality of solutions designed for the assignments, the performance in VIVA-VOCE, the quality of lab. File and ethical practices followed. The internal examination is conducted by the course coordinator. The external examination is conducted by external examiner appointed by the Controller of Examination in association with the internal examiner appointed by the Chairperson of the Department.

Pre-requisites: Basic programming skills

About the Course:

Python is a scripting programming language known for both its simplicity and wide breadth of applications. For this reason it is considered one of the best languages for beginners. Used for everything from web development to scientific computing Python is referred to as a general purpose language by the greater programming community. The major objective of Python language is to make the students solve real word problem efficiently using python library.

Course Outcomes: By the end of the course students will be able to:

CO1. **Implement** solutions to the given assignments in Python. (LOTS: Level 3: Apply)

CO2. **use** various Python packages for solving different programming problems. (LOTS: Level 3: Apply)

CO3. **Devise** solutions for complex problems of data analysis and machine learning. (HOTS: Level 6: Create)

CO4. **Evaluate** the output of data analysis and machine learning models. (HOTS: Level 5: Evaluate)

CO5. **Create** lab records of the solutions for the given assignments. (HOTS: Level 6: Create)

CO6. **Demonstrate** use of ethical practices, self-learning and team spirit.. (LOTS: Level 3: Apply)

List of experiments/assignments

List of experiments/assignments

1. Install Python and explore various popular IDE like IDLE, PyCharm, and Anaconda.

2. Assignments to perform various number operations like

a. Find maximum from a list of numbers

b. GCD of two numbers

c. Square root of a number

d. Check number is prime or not.

e. Print first N prime numbers

f. Remove duplicate numbers from list

g. Print the Fibonacci series.

3. Assignments to perform various operations on Strings like creation, deletion, concatenation.

4. Create a List L = [10, 20, 30]. Write programs to perform following operations:

a. Insert new numbers to list L. b. Delete numbers from list L.

c. Sum all numbers in list L. d. Sum all prime numbers in list L.

e. Delete the list L.

5. Create a Dictionary D= {'Name': 'Allen', 'Age': 27, 5:123456}. Write programs to perform following operations:

a. Insert new entry in D. b. Delete an entry from D.

c. Check whether a key present in D. d. Update the value of a key.

e. Clear dictionary D.

6. Two assignments on Sets to perform various operation like union, intersection, difference etc.

7. Two assignments related to searching operation like linear search, binary search.

8. Three assignments related to sorting like selection sort, bubble sort, insertion sort.

9. Demonstrate the use of dictionary for measuring student marks in five subjects and you have to find the student having maximum and minimum average marks.

10. Two assignment on usage of different available packages like random package to perform

a. Print N random numbers ranging from 100 to 500.

b. Print 10 random strings whose length between 3 and 5.

11. Two assignments on usage of package such as Numpy, Pandas.

12. Implement and demonstrate the functions of a simple calculator.

13. One assignment on implementing object oriented concept such as classes, inheritance, and polymorphism.

14. One assignment on file handling that how data is read and written to a file.

Reference Books:

1. Allen B. Downey , " Think Python: How to Think Like a Computer Scientist", Second Edition, Updated for Python 3, Shroff/O'Reilly Publishers, 2016.
2. Shroff "Learning Python: Powerful Object-Oriented Programming; Fifth edition, 2013.
3. David M.Baezly "Python Essential Reference". Addison-Wesley Professional; Fourth edition, 2009.
4. David M. Baezly "Python Cookbook" O'Reilly Media; Third edition (June 1, 2013) by.
5. http://www.edx.org

Note:

The actual experiments/assignments will be designed by the course coordinator. One assignment should be designed to be done in groups of two or three students. The assignments must meet the objective of the course and the levels of the given course outcomes. The list of assignments and schedule of submission will be prepared by the course coordinator at the beginning of the semester.

CO-PO Articulation Matrix:Python Programming Lab. (PC/AIML/10-P)

List of Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
CO1. **Implement** solutions to the given assignments in Python. (LOTS: Level 3: Apply)	2	1	_	_	3	_	_	_	_	_	_	_	_	3
CO2. **Use** various Python packages for solving different programming problems. (LOTS: Level 3: Apply)	2	3	_	3	3	_	_	_	_	_	_	_	_	3
CO3. **Devise** solutions for complex problems of data analysis and machine learning. (HOTS: Level 6: Create)	3	3	1	3	3	_	_	_	_	_	_	_	_	3
CO4. **Evaluate** the output of data analysis and machine learning models. (HOTS: Level 5: Evaluate)	3	3		3	3	_	_	_	_	_	_	_	_	3
CO5. **Create** lab records of the solutions for the given assignments. (HOTS: Level 6: Create)	_	_	_	_	_	_	_	_	_	3	_	_	_	_
CO6. **Demonstrate** use of ethical practices, self-learning and team spirit. (LOTS: Level 3: Apply)	_	_	_	_	_	_	_	3	3	_	_	3	_	_
Level of Attainments PC/AIML/10-P														

INDUSTRIAL TRAINING/INTERNSHIP

General Course Information

Course Code: EEC/AIML/1 Course Credits: 2 Mode: Industrial Training / Internship	**Course Assessment Methods (100 Marks)** An internal evaluation is done by a faculty member appointed by the Chairperson of the Department. Significance and originality of the problem addressed and the solution provided: 20 Knowledge of the problem domain and tool used (VIVA-VOCE):25 Report Writing: 20 Judgement of the skill learnt and system developed: 20 Level of ethics followed: 15

About the Industrial training:

Students do an Industrial Training of 4 to 6 weeks after fourth semester. They are expected to learn novel skills and develop some software application during the training period.

After doing training students will be able to:

CO1. **Address** novel problems in an original manner using latest skills. (HOTS: Level 6: Create)

CO2. **Select and apply** modern engineering tools. (LOTS: Level 3: Apply)

CO3. **Prepare** training report by organising ideas in an effective manner. (HOTS: Level 6: Create)

CO4. **Engage** in lifelong learning. (HOTS: Level 6: Create)

CO5. **Apply** ethical practices while doing the training and writing report. (LOTS: Level 3: Apply)

CO-PO Articulation Matrix:Industrial Training (INT-CSEAI301)

List of Course Outcomes	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
CO1. **Address** novel problems in an original manner using latest skills (HOTS: Level 6: Create)	_	3	3	2		1	_	_	2	_	1	_	3	3
CO2. **Select and apply** modern engineering tools. (LOTS: Level 3: Apply)	2	–	–	–	3	_	_	_	3	_	_	_	3	2
CO3. **Prepare** training report by organising ideas in an effective manner. (HOTS: Level 6: Create)	–	–	–	–	–	–	–	–	–	–	–	–	–	–
CO4. **Engage** in lifelong learning. (HOTS: Level 6: Create)	_	_	_	_	_	_	_	_	_	_	_	3	_	_
CO5. **Apply** ethical practices while doing the training and writing report. (LOTS: Level 3: Apply	_	_	_	_	_	_	_	3	_	3	_	_	_	_
Level of Attainments INT-CSEAI301														