NEW SCHEME FOR M. Sc. (Artificial Intelligence and Machine Learning) COURSE

SEMESTER - I (M.Sc. (AI & ML)) Applicable from July 2019 onwards

			TEACHING SCHEME			EXAMINATION SCHEME							
SR.	SUBJECT	NAME OF THE SUBJECT	THEORY	TUTO	PRACT.	SESSIONAL		THEORY		PRACT.		T.W.	TOTAL
NO.	NO.		HR	HR.	HR.	M.	HR.	M.	HR	M	H	MARKS	
1	MSCAI 111	Mathematical Foundation	3	1	-	25	2	100	3	-	-	25	150
2	MSCAI 112	Problem Solving With Python	4	-	3	25	2	100	3	50	3	25	200
3.	MSCAI 113	Artificial Intelligence	3	1	0	25	2	100	3	-	-	25	150
4	MSCAI 114	Object Oriented Concepts & Programming using C++	4	-	3	25	2	100	3	50	3	25	200
	MSCAI 115	Linear Algebra and Numerical Methods	4	-	3	25	2	100	3	50	3	25	200
	MSCAI 116	Project - I	-	3	6	25	-	-	-	100	-	25	150
	TOTAL		18	5	15	150	-	500	-	250	-	150	1050

Gujarat University
Syllabus
M.Sc. (Artificial Intelligence and Machine Learning) - I

Course Name: Mathematical Foundation

Course Code: MSCAI 111

Objectives:

With the current deployment of computer technology and tools, it is very important to develop the student's geometric insight into the concepts of Calculus, Vectors and Vector Spaces and applying these concepts to real life problems and machine learning problems
Aim of the course is to enable students

- To introduce the Concepts of Calculus, Vectors and Vector Spaces
- To apply these concepts to real life problems and machine learning problems

Prerequisites:

Basic knowledge of Mathematical fundamentals

Contents:

1. **Introduction to Set Theory**
   Basic Concepts, notations, inclusion and equality, power set, operations of union, intersection and complement, Venn diagrams, set identities, ordered pairs and n-tuples, Cartesian product

2. **Introduction to Coordinate Geometry**
   Definition of coordinates and axes, coordinate plane, plotting of points, scatter diagram, general form of a straight line, slope and intercept, distance formula, section formula, mid-point formula angle between two lines, triangle in Cartesian plane, distance of a point from the line, equation of a normal to the line, Equation of plane, hyper plane, equation of normal to plane, to classify point to side of plane

3. **Fundamentals of Single Variable Calculus**
   Functions of single variable, definition and their graphs, special functions like polynomials, trigonometric , exponential , hyperbolic , limit, continuity, definition of derivative and its graphical meaning , rules of differentiation, chain rule, higher order derivatives, definition of integration and its geometric interpretation ,indefinite and definite integral and their evaluation, Optimization of functions : Local Maxima and minima of functions, saddle point, , necessary and sufficient conditions, global maxima, convex functions, Taylor Series

4. **Multivariable Calculus**
   Multivariable functions and their 3 D graphs, contour lines and maps, introduction to partial derivatives and formal definition, graphical meaning, computing of partial derivatives, chain rule, second order partial derivatives and their symmetry, higher order partial derivatives, Gradient and its physical interpretation, properties, directional derivatives, Jacobian, computing Jacobian matrix and its determinant, Lagrange multiplier method for finding local optimum

5. **Fundamentals of Vectors**
   Definition of vector, scalars, addition and subtraction of vectors, scalar multiplication, inner product(dot product) of vectors, norms, direction, orthogonal vectors, projection of vectors, cosine similarity, normal and orthonormal vectors, Gram-Schmidt procedure, orthogonal decomposition

6. **Vector Spaces**
   Vector Space and Subspaces, null Space, row space and column space , linear transformations, linear independence, basis vectors , linear combination, dimension , linear span, change of basis, invariant subspaces

Main Reference Books:

1. Advanced Engineering Mathematics, 10ed, ISV, Erwin Kreyszig, John Wiley & Sons, Inc.
2. Linear Algebra and Its Application, 3rd Edition, David C. Lay, Pearson
3. Introduction to Algorithms, 3rd Ed, Corman, Leiserson, Rivest and Stein, PHI
4. Advanced Mathematics for Engineers, Dr. Chandrika Prasad, Pothishala Private Ltd.
5. Linear Algebra, Kenneth Hoffman, PHI
6. Linear Algebra and Its Applications, Gilbert Strang, Cambridge University Press

Accomplishments of the student after completing the Course:

- Demonstrate the ability to apply mathematical skills for problem solving applications
- Apply mathematical techniques of geometry, calculus and vectors to solve problems
- Represent and evaluate basic mathematical and/or logical information numerically, graphically, and symbolically
- Interpret mathematical and/or logical models such as single and multivariable calculus as well as vectors and vector space

***** ***** *****

Course Name: Problem Solving with Python

Course Code: MSCAI 112

Objectives:

Introduce students to Python, the modern language useful for writing compact codes specifically focusing on Data Analysis and Scientific Computing. Equal weightage has been given to both theory and practical

Aim of the course is to enable students

- To introduce the principles of Python Programming
- To understand and use functionality of various Python libraries for Network Programming
- To gain basic insight of programming that can be used over Machine Learning Deep and Learning for problem solving

Prerequisites:

Fundamentals of Computers

Contents:

1. **Introduction**
   Introduction to Python, Data types in Python, Built in data type, Bool data type , Sequences in Python, Sets, Literals in python, Operator in Python, Arrays in Python, Strings and Characters, Control structure – Condition execution in Python, Using iteration within Python programs, Arrays in Python, Strings and characters in Python

2. **Functions in Python**
   Difference between a Function and a Method, Defining a Function, Calling a Functions, Returning Results from a Function, Returning Multiple Values from a Function, Functions are First Class Objects, Pass by Object reference, Formal and Actual Arguments, Positional Arguments, Keyword Arguments, Default Arguments, Variable Length Arguments, Local and Global Variables, The Global Keyword, Passing a Group of Elements to a function, Recursive Functions, Anonymous Functions or Lambdas, Function Decorators, Generators, Structured Programming, Creating our Own Modules in Python

3. **Lists, Tuples, Dictionaries & Sets in Python**
   Lists: Using Lists, List Traversal, Building Lists, List Membership, List Assignment and Equivalence, List Bounds, Slicing, List Element Removal Lists and Functions, List Methods, Prime Generation with a List, Command-line Arguments, List Comprehensions,

Multidimensional Lists, Lists Vs. Generators

Tuples, Dictionaries, Sets: Tuples, Arbitrary Argument Lists, Dictionaries, Using Dictionaries, Counting with Dictionaries, Grouping with Dictionaries, Keyword Arguments, Sets, Set Quantification with all and any, enumerating the Elements of a Data Structure

4. **Object Oriented Programming in Python**

   Introduction to Object Oriented Programming, Classes and Objects, Inheritance and Polymorphism, Abstract Classes and Interfaces, Exceptions

5. **Files in Python**

   Files, Types of Files in Python, Opening a File, Closing a File, Working with Text Files Containing Strings, Knowing Whether a File Exists or Not, Working with Binary Files, The with Statement, Pickle in Python, The seek() and tell() Methods, Random Accessing of Binary Files, Random Accessing of Binary Files Zipping and Unzipping Files, Working with Directories, Running Other Programs from Python Program, Points to Remember

Main Reference Books:

1. Core Python Programming, Rao N.R., Dreamtech Publication India
2. Foundations of Python Networking, Rhodes & Goerzen, Apress Publication
3. Python Network Programming Cookbook, Sarker M.O.F., Packt Publication
4. Fundamentals of Python Programming, Halterman R., Southern Adventist University
5. Introduction to Computation and Programming Using Python, Guttag J.V., Prentice Hall India
6. Core Python Programming, Chun W., Prentice Hall India

Accomplishments of the student after completing the Course:

- Be able to Solve challenging problems using Python programming language

***** ***** *****

Course Name: Introduction to Artificial Intelligence

Course Code: MSCAI 113

Objectives:

Introduce and define the meaning of Intelligence and explore various paradigms for knowledge encoding in computer systems is the basis of this course. Also it is important to introduce subfields of AI such as NLP, Game Playing, Bayesian Models, etc.

Aim of the course is to enable students to

- Learn and understand the concepts of artificial intelligence
- Understand the concepts of NLP, Bayesian Models and Game playing

Prerequisites:

Basic knowledge of Mathematical Logic

Contents:

1. **Introduction to AI fundamentals**
   Defining Artificial Intelligence, History of AI, AI task domains, Defining AI techniques, Turing Test, Intelligent Agents: Agents and Environments, Nature of Environments, Rationality, Performance Measures, Structure of Agents, Problem-Solving Agents, Knowledge-Based Agents

2. **State Space Search and Heuristic Search Techniques**
   Defining problems as state space search, problem characteristics, production systems and characteristics, heuristics, breadth first and depth first search, Heuristic search, Best first search, Hill climbing, problems of Hill climbing techniques, A*, AND-OR graphs, iterative deepening A*

3. **Representing Knowledge**
   Knowledge Representation Techniques: Computable functions and predicates, Backward Chaining, Procedural vs. Declarative Knowledge, Forward vs. Backward Reasoning, Semantic Networks, Partitioned Semantic Networks, Conceptual Dependency, Issues in Knowledge Representation

4. **Symbolic Logic under Uncertainty**
   Uncertainty, Combining Beliefs and Desires under Uncertainty, Basis of Utility Theory, Utility Functions, Decision Networks, Monotonic vs. Non-monotonic Reasoning

5. **Game Playing**
   Games, optimal decisions in Games, Minimax method and its complexity, perfect and imperfect decisions, Alpha Beta pruning and its effectiveness, other refinements

6. **Expert Systems**
   Introduction to Expert systems, knowledge representation, architecture and knowledge engineering , Rules as knowledge representation technique, characteristics, Advantages and disadvantages, Forward and Backward Chaining, Real time example of Rule-based expert system

Main Reference Books:

1. Artificial Intelligence-A modern Approach, Stuart Russell and Peter Norvig, 3rd Edition, Pearson Education
2. Artificial Intelligence, Elaine Rich, Kevin Knight, Shivashankar B. Nair, 3rd Edition, McGraw Hill
3. Artificial Intelligence-Structures and Strategies for Complex Problem Solving, George F. Lugar, 5th Edition, Pearson Education
4. Artificial Intelligence-A guide to Intelligent Systems, Michael Negnevitsky, 2nd Edition, Pearson Education
5. Expert Systems-Principals and Programming, Joseph C. Giarratano, 4th Edition, Pearson Education
6. Artificial Intelligence and Intelligent Systems, N.P. Padhy, Oxford University Press

Accomplishments of the student after completing the Course:

- Store and represent the knowledge in various applications and use different AI searching techniques
- Deal with poorly defined or inexact problems that do not respond to the algorithmic solutions

***** ***** *****

Course Name: Object Oriented Concepts and Programming using C++

Course Code: MSCAI 114

Objectives:

The programming for small devices like mobile phones, networking devices like routers, coding for graphics and multimedia, requires efficient coding as well as object oriented programming. The C++ language fits perfectly as a tool for this type of work. How this important language is to be mastered and how to use this knowledge in building efficient and flexible code is one of the prime requirements today. The course presented here is targeting to enable the student to master such skills.

Aim of the course is to enable students to

- Differentiate between procedural and object oriented programming
- Learn C++ as a language and various features of it
- Learn Object Oriented principles and their application using C++

Prerequisites:

Knowledge of C language, Programming concepts including algorithm building and logical problem solving

Contents:

1. **Introduction to C++, Overview of Core C++ Language, Classes and Objects**
   Identifiers and constants (Literals), Keywords, Data Types, The Operators, New Casting Operators, typeid and throw, The Conditional structures and Looping Constructs, , The Difference between struct and class in C++,The difference between Union and Class, Static Data members of a class, Pointer to objects and pointer to members of class, The local classes, Assigning Objects

2. **Functions**
   Introduction, The inline function, Default Arguments to the function,Functions with object as parameters,Call by reference and return by reference, Prototyping and Overloading, Friend functions, Const and Volatile functions, Static functions, Private and Public functions, Function Pointers, Adding C functions to the C++ program

3. **Constructors and Destructors**
   Introduction to constructors, The explicit constructors, Parameterized constructors, Having multiple constructors, Constructors with default arguments, Dynamic Initialization, Constructor with dynamic allocation, copy constructors, The member initialization list,

destructors

4. **Operator Overloading and User Defined Conversions**
   Introduction, Unary Operators, Binary Operators, Using Friends as operator functions, Overloading other Operators, The need for user defined conversion, Four different cases where user defined conversions are needed, Comparison of both the methods of conversion

5. **Templates**
   Function Templates, Non Generic (Non Type) Parameters in Template functions, Template function and specialization, Overloading a template function, Using Default Arguments, Class Templates, Classes with multiple generic data types, Static data members, Primary and Partial Specialization, The Export Keyword, The other use of typename

6. **Inheritance**
   The need, Defining derived class using single base class, Derivation using public, private and protected access modifiers, The implementation of inheritance in the C++ object model, The Access Control, The Access Declaration, The multiple-inheritance, Abstract classes, Composite objects (container objects)

7. **Runtime polymorphism by virtual functions**
   Compile Time and Runtime Polymorphism, Pointers to Objects, This pointer, Compatibility of Derived and base class pointers, The subobject concept, Virtual functions, Static invocation of virtual function, Default arguments to virtual functions, Virtual destructors, Pure virtual functions

8. **IO Streams**
   Need for streams, Advantages of using C++ I/O over C IO, The C++ Predefined streams, Formatting IO, Formatting using ios members, Manipulators, Creating our own manipulator

9. **Using Files for IO**
   Why IO is special, Text and binary streams, Opening and closing files, Dealing with text files Dealing with binary files, Providing Random Access using seek, IO Modes, Handling Errors

10. **Namespaces**
    Introduction and need, Use the using syntax, Defining namespaces, Extending the namespace, Unnamed namespaces, Nested Namespaces, Namespace aliases, The std namespace, The Koenig lookup, Overhead with namespaces

11. **The Standard Template Library**

The STL (Standard Template Library) Introduction, Generic Programming, Generic Software Components, Generic Algorithms, Iterators, Containers, Algorithms

Main Reference Books:

1. Programming with ANSI C++, Bhushan Trivedi, Oxford University Press
2. C++ Primer, Stanley Lippmann Pearson Education
3. The C++ Programming Language, Bjarne Stroustrup, Pearson Education
4. Effective C++, Scott Mayer Addison Wesley
5. Complete Reference C++, Herbert Schildt McGraw Hill Publications
6. C++ FAQs, Pearson Education

Accomplishments of the student after completing the Course:

- Understand and appreciate the Object Oriented approach of programming
- Being aware of the working and architectural model of C++
- Able to solve problems given to him/her using C++ keeping balance between efficiency and flexibility

***** ***** *****

Course Name: Linear Algebra and Numerical Methods

Course Code: MSCAI 115

Objectives:

To review, strengthen and teach important Mathematical concepts required for AI & ML which the student has already been either exposed to in previous programs or is required to study
Aim of the course is to enable students to

- Learn and implement Mathematical concepts which are required in this course

Prerequisites:

Basic knowledge of Mathematical fundamentals

Contents:

1. **Introduction to Matrices**
   Definition, addition of two matrices, transpose, scalar multiplication, matrix multiplication, properties of matrix addition and multiplication, square matrix , null and identity matrix , invertible matrix and inverse, hadamard product and its properties, determinant of a square matrix and its properties, rank , trace, popular type of matrices-symmetric, diagonal, orthogonal, orthonormal, positive definite matrix

2. **Introduction to Numerical Methods**
   Introduction, Characteristic of numerical methods, Types and sources of errors in data, Quantification of errors, nature of iterative methods to find a solution, numerical methods of finding roots of an equation f(x) = 0 : Bisection method, false position method, Secant method, Newton Raphson Method , Gradient Descent method

3. **Linear Equations**
   Systems of Linear Equations, Cramer's Rule, Elementary row operations, row reduced and Echelon forms, Homogeneous Systems, Matrix inversion method, Gaussian Elimination method, Ill-Conditioned Systems , Iterative methods Gauss-Jordon Method , Gauss Seidel Method

4. **Eigen Values and Eigen Vectors**
   Eigen Values and Eigen Vectors, Characteristic Equation of a matrix, Properties of Eigenvalues and Eigenvectors, Iterative methods to determine eigen values, Diagonalization of symmetric matrices, Orthogonal Diagonization , Singular Value Decomposition, Principal Component Analysis

5. **Computational Complexity**
   Time Complexity: Growth of functions, Asymptotic notations, NP-Completeness and the P & NP Classes: Introduction, Polynomial Time & Verification, NP-Completeness and Reducibility, The Vertex Cover Problem ,The Traveling Salesman Problem, The Set Covering Problem

Main Reference Books:

1. Numerical Methods for Engineers, Steven C Chapra & Raymond P Canale, Fifth Edition, Tata McGraw Hill Publication, Special Indian Edition
2. Applied Numerical Analysis, C F Gerald & P O Wheatley, Seventh Edition, Pearson Education Asia, New Delhi
3. Linear Algebra and Its Application, 3rd Edition, David C. Lay, Pearson
4. Linear Algebra, Kenneth Hoffman, PHI
5. Linear Algebra and Its Applications, Gilbert Strang, Cambridge University Press
6. Numerical Methods, Dr. V. N. Vedamurthy & Dr. N.Ch. S.N. Iyengar, Vikas Publication

Accomplishments of the student after completing the Course:

- Be able to apply properties and solutions to system of linear equations, matrices, properties of matrices, matrix algebra, determinants, eigenvalues, real vectors in two and three dimensions, vector algebra (including dot and cross products), linear combinations, and linear independence

***** ***** *****