

DEPARTMENT OF INFORMATION SCIENCE AND ENGINEERING

SCHEME OF TEACHING AND EXAMINATION (2017 – 2018)

B.E. (ISE) VII SEMESTER

Sl. No	Subject Code	Subject	Credits	Hours/Week			Examination Marks		
				Lecture	Tutorial	Practical	CIE	SEE	Total
1	UIS707C	Unix System Programming	04	4	0	0	50	50	100
2	UIS702C	Object Oriented Modeling and Design	04	4	0	0	50	50	100
3	UIS708H	Software Project Management	03	3	0	0	50	50	100
4	UIS056E	Network Management System	04	4	0	0	50	50	100
5	UIS057E	C# Programming. NET	04	4	0	0	50	50	100
6	UIS060E	Machine Learning	04	4	0	0	50	50	100
7	UIS727L	Object Oriented System Design Lab	1.5	0	0	3	50	50	100
8	UIS712L	Unix System Programming Lab	1.5	0	0	3	50	50	100
9	UIS711P	Project Phase - I	03	0	0	3	50	50	100
		Total	25	19	00	9	400	400	800

UIS707C: Unix System Programming
(4-0-0)(4 Credits, 52 Hours)

- Course Objectives** :
- 1) To understand the fundamental design of the unix operating system.
 - 2) To become fluent with the systems calls provided in the unix environment.
 - 3) To be able to design and build an application/service over the unix operating system.
 - 4) To be able to use operating system kernel calls within a programming language to interact with the file system, set and respond to timers/interrupts.

- Course Outcomes** :
- 1) Comprehend the UNIX and ANSI standards ,UNIX and POSIX API's and demonstrate the ability to write C/C++ programs using facilities offered by the above standards and file API's.
 - 2) Exhibit the ability to comprehend the UNIX kernel support for process and demonstrate C/C++ program using process API's.
 - 3) Illustrate the ability to demonstrate C/C++ program using signal API's.
 - 4) Comprehend the different IPC method (Message Queue, Semaphore, and Shared Memory) and develop C/C++ program using IPC related API's
 - 5) Exhibit the ability to develop C/C++ program using socket API's for client/server module (UDP,TCP).
 - 6) Exhibit the ability to develop C/C++ program using TLI API's.

Mapping (Co-relations) of CO's with PO's

Sl. No.	Course Outcomes	IS-PO-1	IS-PO-2	IS-PO-3	IS-PO-4	IS-PO-5	IS-PO-6	IS-PO-7	IS-PO-8	IS-PO-9	IS-PO-10	IS-PO-11	IS-PO-12
01	Comprehend the UNIX and ANSI standards ,UNIX and POSIX API's and demonstrate the ability to write C/C++ programs using facilities offered by the above standards and file API's.	3	3		3	1			2				
02	Exhibit the ability to comprehend the UNIX kernel support for process and demonstrate C/C++ program					2	3				1		

	using process API's.												
03	Illustrate the ability to demonstrate C/C++ program using signal API's.	3								3			
04	Comprehend the different IPC method (Message Queue, Semaphore, and Shared Memory) and develop C/C++ program using IPC related API's.	2									3		
05	Exhibit the ability to develop C/C++ program using socket API's for client/server module (UDP,TCP).								3		3		
06	Exhibit the ability to develop C/C++ program using TLI API's.			3							2		

Syllabus

Academic Year	:	2018 - 19
Semester	:	VII
Subject	:	Unix System Programming
Subject Code	:	UIS707C
Credits	:	04
Teaching Hours	:	52
Unit –I		13 Hours
UNIX and ANSI Standards:		
The ANSI C Standard, The ANSI/ISO C++ Standards, Differences between ANSI C and C++, The POSIX Standard, The POSIX.1 FIPS Standard, The X/Open Standards.		
UNIX and POSIX APIs:		
The POSIX APIs, The UNIX and POSIX Development Environment, API Common Characteristics.		
UNIX Files:		
File Types, The UNIX and POSIX File system, The UNIX and POSIX File Attributes, Inodes in UNIX System V, UNIX Kernel Support for Files, Hard and Symbolic links.		
UNIX File API'S:		
General File API's, File and record locking, Directory file API's.		
Unit –II		13 Hours
UNIX PROCESSES		

<p>Unix Kernel Support for Processes, Process API's, Process Attributes, Change Process Attributes.</p> <p>UNIX SIGNALS</p> <p>The Unix Kernel Support for Signals, signal, Signal mask, sigaction, the SIGCHLD Signal and the waitpid function, The sigsetjmp and siglongjmp Functions, kill alarm, Interval Timers.</p>		
Unit -III		13 Hours
<p>INTERPROCESS COMMUNICATION</p> <p>POSIX.1b IPC methods, The UNIX system V IPC methods, Unix system V messages, POSIX.1b messages, Unix system V semaphores, Unix system V shared memory, Memory mapped I/O.</p>		
Unit-IV		13 Hours
<p>SOCKETS and TLI</p> <p>Sockets, A stream socket example, Client –server message handling example, TLI, TLI class, Client server message example, Datagram example.</p>		
Text Book	:	1) Terrence. Chan, “ Unix System Programming Using C++ ”, Printice Hall India.
Reference Books	:	1) W. Richard. Stevens, “ Advanced Programming in the Unix Environment ”, Addison Wesley/PHI.

UIS702C: Object Oriented Modeling and Design
(4-0-0)(4 Credits, 52 Hours)

- Course Objectives** :
- 1) To understand the object oriented concepts for designing object oriented models.
 - 2) To understand the use of UML (Unified Modeling Language) for object oriented analysis and design.
 - 3) To understand the problems, communicating with application experts, modeling enterprises, preparing documentation, and designing programs by using object oriented models.
 - 4) To describe the step by step object oriented methodology of software development from problem statement through analysis, system design, and class design.
 - 5) To understand the issues for implementing object oriented designs or models.
 - 6) To understand the concept of different patterns for constructing software architectures through object oriented models.

- Course Outcomes** :
- 1) Demonstrate the ability to apply the knowledge of object orientation in modeling and design.
 - 2) Demonstrate the ability to apply the concept of structural and behavioral models using UML appropriate notations.
 - 3) Describe a step by step object oriented methodology of software development from problem statement through analysis, system design, and class design.
 - 4) Comprehend the implementation of object oriented designs in object oriented languages.
 - 5) Illustrate the concept of patterns for constructing software architectures.

Mapping (Co-relations) of CO's with PO's

Sl. No.	Course Outcomes	IS-PO-1	IS-PO-2	IS-PO-3	IS-PO-4	IS-PO-5	IS-PO-6	IS-PO-7	IS-PO-8	IS-PO-9	IS-PO-10	IS-PO-11	IS-PO-12
01	Demonstrate the ability to apply the knowledge of object orientation in modeling and design.	-	-	-	3	2	-	-	-	-	2	3	2
02	Demonstrate the ability to apply the concept of structural and behavioral models using UML appropriate notations.	-	-	-	2	3	-	-	-	-	3	2	1
03	Describe a step by step object oriented methodology of software development from problem statement through analysis, system design, and class design.	-	-	-	2	3	-	-	-	-	3	3	2
04	Comprehend the implementation of object oriented designs in object oriented languages.	-	-	-	2	3	-	-	-	-	3	3	2
05	Illustrate the concept of patterns for constructing software architectures.	-	-	-	1	2	-	-	-	-	2	2	1

Syllabus

Academic Year	:	2018 - 19
Semester	:	VII
Subject	:	Object Oriented Modeling and Design
Subject Code	:	UIS702C
Credits	:	04
Teaching Hours	:	52 Hours
Unit –I		13 Hours
<p>INTRODUCTION, MODELING CONCEPTS, CLASS MODELING: Introduction to Object Orientation, OO development, OO themes; Evidence for usefulness of OO development; OO modeling history. Modeling as Design Technique: Modeling; abstraction; the three models. Class Modeling: Object and class concepts; Link and associations concepts; Generalization and inheritance; A sample class model; Navigation of class models; Practical tips. Advanced Class Modeling: Advanced object and class concepts; Association ends; N-ary associations; Aggregation; Abstract classes; Multiple inheritance; Metadata; Reification; Constraints; Derived data; Packages; Practical tips.</p>		
Unit –II		13 Hours
<p>STATE MODELING, ADVANCED STATE MODELING, INTERACTION MODELING, and PROCESS OVERVIEW: State Modeling: Events, States, Transitions and Conditions; State diagrams; State diagram behavior; Practical tips. Advanced State Modeling: Nested state diagrams; Nested states; Signal generalization; Concurrency; A sample state model; Relation of class and state models; Practical tips. Interaction Modeling: Use case models; Sequence models; Activity models. Use case relationships; Procedural sequence models; Special constructs for activity models. Process Overview: Development stages; Development life cycle.</p>		
Unit -III		13 Hours
<p>SYSTEM CONCEPTION, DOMAIN ANALYSIS, APPLICATION ANALYSIS, and SYSTEM DESIGN - 1: System Conception: Devising a system concept; Elaborating a concept; Preparing a problem statement. Domain Analysis: Overview of analysis; Domain class model; Domain state model; Domain interaction model; Iterating the analysis. Application Analysis: Application interaction model; Application class model; Application state model; Adding operations. System Design -1: Overview of system design; Estimating performance; Making a reuse plan; Breaking a system in to sub-systems; Identifying concurrency; Allocation of sub-systems; Management of data storage; Handling global resources; Choosing a software control strategy.</p>		
Unit-IV		13 Hours
<p>SYSTEM DESIGN - 2, CLASS DESIGN, IMPLEMENTATION MODELING AND DESIGN PATTERNS: System Design -2: Handling boundary conditions; Setting the trade-off priorities; Common architectural styles; Architecture of the ATM system as the example. Class Design: Overview of class design; Bridging the gap; Realizing use cases; Designing algorithms; Recursing downwards, Refactoring; Design optimization; Reification of behavior; Adjustment of inheritance; Organizing a class design; ATM example. Implementation Modeling: Overview of</p>		

implementation; Fine-tuning classes; Fine-tuning generalizations; Realizing associations; Testing. **Design patterns:** What is a pattern and what makes a pattern? Pattern categories; Relationships between patterns; Pattern description. Communication Patterns: Forwarder-Receiver; Client-Dispatcher-Server; Publisher-Subscriber.

Text Books	:	<ol style="list-style-type: none"> 1) Michael. Blaha, James. Rumbaugh “Object-Oriented Modeling and Design with UML”, 2nd Edition, Pearson Education, 2005. 2) Frank. Buschmann, Regine. Meunier, Hans. Rohnert, Peter. Sommerlad, Michael. Stal, “Pattern Oriented Software Architecture A System of Patterns”, Volume.1, John Wiley and Sons, 2008.
Reference Books	:	<ol style="list-style-type: none"> 1) Ali. Bahrami, “Object Oriented Systems Development”, McGraw-Hill, 2008. 2) Grady. Booch “Object-Oriented Analysis and Design with Applications”, 3rd Edition, Pearson, 2007. 3) Mark. Priestley, “Practical Object-Oriented Design with UML”, 2nd Edition, Tata McGraw-Hill, 2003.

UIS708H: Software Project Management
(3-0-0)(3 Credits, 40 Hours)

- Course Objectives** :
1. Selection and initiation of individual projects in the enterprise.
 2. Conduct project planning activities that forecast project costs and timelines.
 3. Implement processes for successful resource, communication, risk and change management.
 4. Demonstrate effective project execution and control techniques that result in successful projects.
 5. Conduct project activities and obtain formal project acceptance.
 6. Demonstrate effective organizational leadership and change skills for managing projects, project teams, and stakeholders.

- Course Outcomes** :
1. Identify Software projects, their stages and planning.
 2. Evaluate and select projects against Strategic, Technical and Economic criteria.
 3. Estimate an activity plan, resource schedule and risk factors with their effects for a project.
 4. Monitor the progress of the project and controlling changes to a projects requirement.
 5. Administer a contract from signing to acceptance and identifying the right Human resources and their best utilization.

Mapping (Co-relations) of CO's with PO's

Sl. No.	Course Outcomes	IS-PO-1	IS-PO-2	IS-PO-3	IS-PO-4	IS-PO-5	IS-PO-6	IS-PO-7	IS-PO-8	IS-PO-9	IS-PO-10	IS-PO-11	IS-PO-12
01	Identify Software projects, their stages and planning.	--	--	--	3	--	--	3	1	1	3	2	--
02	Evaluate and select projects against Strategic, Technical and Economic criteria.	--	--	--	1	3	--	2	1	--	3	3	--
03	Estimate an activity plan, resource schedule, risk factors with their effects for a project.	--	--	--	2	2	--	--	1	--	1	2	3
04	Monitor the progress of the project and controlling changes to a projects requirement.	--	--	--	--	1	--	--	1	--	--	2	3
05	Administer a contract from signing to acceptance and identifying the right Human resources and their best utilization.	--	--	--	--	--	2	--	1	3	--	3	3

Syllabus

Academic Year	:	2018 - 19
Semester	:	VII
Subject	:	Software Project Management
Subject Code	:	UIS708H
Credits	:	03
Teaching Hours	:	03
Unit –I		10 Hours
INTRODUCTION TO SOFTWARE PROJECT MANAGEMENT Project Definition – Contract Management – Activities Covered By Software Project Management – Overview Of Project Planning – Stepwise Project Planning.		
Unit –II		10 Hours
PROJECT EVALUATION: Strategic Assessment – Technical Assessment – Cost Benefit Analysis – Cash Flow Forecasting – Cost Benefit Evaluation Techniques – Risk Evaluation.		
Unit -III		10 Hours
ACTIVITY PLANNING: Objectives – Project Schedule – Sequencing and Scheduling Activities Network Planning Models – Forward Pass – Backward Pass – Activity Float – Shortening Project Duration – Activity on Arrow Networks – Risk identification – Assessment – Monitoring – PERT technique – Monte Carlo simulation – Resource Allocation – Creation of critical patterns – Cost schedules.		
Unit-IV		10 Hours
MONITORING AND CONTROL: Creating Framework – Collecting The Data – Visualizing Progress – Cost Monitoring – Earned Value – Prioritizing Monitoring – Getting Project Back to Target – Change Control – Managing Contracts – Introduction – Types of Contract – Stages in Contract Placement – Typical terms of a Contract – Contract Management – Acceptance. MANAGING PEOPLE: Introduction – Understanding Behavior – Selecting The Right Person For The Job – Instruction In The Best Methods – Motivation – Working In Groups – Becoming A Team –Decision Making		
Text Book	:	1. Bob Hughes, Mike Cotterell: Software Project Management – 6 th Edition, Tata McGraw Hill, New Delhi, 2006.
Reference Books	:	1. Royce, “ Software Project Management ”, Pearson Education, 1999. 2. Jalote, “ Software Project Management in Practice ”, Pearson Education, 2002. 3. Gopaldaswamy Ramesh, “ Managing Global Software Projects ”, McGraw Hill Education (India), Fourteenth Reprint 2

CO 5	To explain management of Ethernet, RMON and broadband networks			3	4						1		
CO 6	To explain the applications of various network management functions						2				1	1	

Syllabus

Academic Year	:	2018–19
Semester	:	VII
Subject	:	Network Management System
Subject Code	:	UIS056E
Credits	:	4
Teaching Hours	:	52
Unit –I		12 Hours
INTRODUCTION, N/W MANAGEMENT STANDARDS, MODELS:		
Introduction: Analogy of Telephone Network Management, Data and Telecommunication Network Distributed computing Environments, TCP/IP Based Networks: The Internet and Intranets, Communications Protocols and Standards- Communication Architectures, Protocol Layers and Services; Case Histories of Networking and Management – The Importance of topology, Filtering Does Not Reduce Load on Node, Some Common Network Problems; Challenges of Information Technology Managers		
Network Management: Goals, Organization, and Functions- Goal of Network Management, Network Provisioning, Network Operations and the NOC, Network Installation and Maintenance; Network and System Management, Network Management System platform, Current Status and Future of Network Management. Network Management Standards, Network Management Model, Organization Model, Information Model – Management Information Trees.		
Unit –II		12 Hours
N/W MANAGEMENT LANGUAGE, SNMPV1 NETWORK MANAGEMENT – 1:		
Communication Model; ASN.1- Terminology, Symbols, and Conventions, Objects and Data Types, Object Names, An Example of ASN.1 from ISO 8824		
Encoding Structure; Macros, Functional Model. Snmpv1 network management – 1: Managed Network: The History of SNMP Management, Internet Organizations and standards, Internet Documents, The SNMP Model, The Organization Model, System Overview.		
Unit -III		12 Hours
SNMPV1 NETWORK MANAGEMENT – 2 SNMP MANAGEMENT – RMON:		
The Information Model – Introduction, The Structure of Management Information, Managed Objects, Management Information Base. The SNMP Communication Model – The SNMP Architecture, Administrative Model, SNMP Specifications, SNMP Operations, SNMP MIB Group, Functional Mode		
SNMP management – RMON: Remote Monitoring, RMON SMI and MIB, RMON1- RMON1 Textual Conventions, RMON1 Groups and Functions, Relationship Between Control and Data Tables, RMON1 Common and Ethernet Groups, RMON Token Ring Extension Groups. RMON2 – The RMON2 Management Information Base, RMON2 Conformance Specifications; ATM Remote Monitoring, A Case Study of Internet Traffic Using RMON.		
Unit-IV		12 Hours

RMON2,BROADBAND N/W MANAGEMENT,N/W MANAGEMENT APPLICATIONS:

Broadband Network Management: ATM Network: Broadband Networks and Services, ATM Technology – Virtual Path-Virtual Circuit, TM Packet Size, Integrated Service, SONET, ATM LAN Emulation, Virtual LAN; ATM Network Management – The ATM Network Reference Model, The Integrated Local Management Interface, The ATM Management Information Base, The Role of SNMP and ILMI in ATM Management, M1 Interface: Management of ATM Network Element, M2 Interface: Management of Private Networks, M3 Interface: Customer Network Management of Public Networks, M4 Interface: Public Network Management, Management of LAN Emulation, ATM Digital Exchange Interface Management
Network Management Applications: Configuration Management- Fault Management, Performance Management, Event Correlation, Security Management, Accounting Management applications overview, Policy- Based Management, Service Level Management.

Text Book	:	Mani Subramanian “Network Management- Principles and Practice”, Pearson Education, 2 nd edition, 2010.
Reference Books	:	J. Richard Burke, “ Network Management Concepts and Practices A Hands-On Approach ”, PHI, 2008.

UIS057E: C# Programming and .NET
(4-0-0)(4 Credits, 52 Hours)

- Course Objectives** :
- 1) To provide the knowledge on core concepts of Building Block of the .NET Platform (CLR, CTS, and CLS).
 - 2) The course also provides the knowledge on C# language fundamentals (creating objects, types, C# syntactic constructs and etc).
 - 3) The course provides the knowledge on Object Oriented Programming in C#, Handling Exceptions and object life time.
 - 4) The course provides the knowledge on Interfaces, Collections, Delegates and Events.

- Course Outcomes** :
- 1) To explain features provided by .NET and execute small programs in console window.
 - 2) To explain the concept of building blocks of .NET (CLR, CTS, CLS).
 - 3) To explain the concept of .dll, binary files and build applications in C#.
 - 4) To explain the fundamentals and object oriented concepts of C# language.
 - 5) To build small programs by using C# language.
 - 6) To build programs using interfaces and collections.

Mapping (Co-relations) of CO's with PO's

Sl. No.	Course Outcomes	IS-PO-1	IS-PO-2	IS-PO-3	IS-PO-4	IS-PO-5	IS-PO-6	IS-PO-7	IS-PO-8	IS-PO-9	IS-PO-10	IS-PO-11	IS-PO-12
01	To explain features provided by .NET and execute small programs in console window.	1	2	2									
02	To explain the concept of building blocks of .NET (CLR, CTS, CLS).		1	1									
03	To explain the concept of .dll, binary files and build applications in C#.			2	1	1							
04	To explain the fundamentals and object oriented concepts of C# language.		2	2									
05	To build small programs by using C# language.			3	3	2							
06	To build programs using interfaces and collections.			3	3	3	1			1			

Syllabus

Academic Year	:	2018 - 19
Semester	:	VII
Subject	:	C# PROGRAMMING AND .NET
Subject Code	:	UIS057E
Credits	:	04
Teaching Hours	:	52
Unit –I		13 Hours
<p>THE PHILOSOPHY OF .NET: Understanding the Previous State of Affairs, The .NET Solution, The Building Block of the .NET Platform (CLR,CTS, and CLS), The Role of the .NET Base Class Libraries, What C# Brings to the Table, An Overview of .NET Binaries (aka Assemblies), Intrinsic CTS Data Types, Understanding the Common Languages Specification, Understanding the Common Language Runtime A tour of the .NET Namespaces, Increasing Your Namespace Nomenclature, Deploying the .NET Runtime, BUILDING C# APPLICATIONS: The Role of the Command Line Compiler (csc.exe), Building C # Application using csc.exe Working with csc.exe Response Files, Generating Bug Reports , Remaining C# Compiler Options, The Command Line Debugger (cordbg.exe) Using the, Visual Studio .NET IDE, Other Key Aspects of the VS.NET IDE, C# “Preprocessor:” Directives, An Interesting Aside: The System .Environment Class.</p>		
Unit –II		13 Hours
<p>C# LANGUAGE FUNDAMENTALS: The Anatomy of a Basic C# Class, Creating objects: Constructor Basics, The Composition of a C# Application, Default Assignment and Variable Scope, The C# Member Initialization Syntax, Basic Input and Output with the Console Class, Understanding Value Types and Reference Types, The Master Node: System, Object, The System Data Types (and C# Aliases), Converting Between Value Types and Reference Types: Boxing and Unboxing, Defining Program Constants, C# Iteration Constructs, C# Controls Flow Constructs, The Complete Set of C# Operators, Defining Custom Class Methods, Understating Static Methods, Methods Parameter Modifies, Array Manipulation in C #, String Manipulation in C#, C# Enumerations, Defining Structures in C#, Defining Custom Namespaces</p>		
Unit -III		13 Hours
<p>OBJECT- ORIENTED PROGRAMMING WITH C#: Forms Defining of the C# Class, Definition the “Default Public Interface” of a Type, Recapping the Pillars of OOP, The First Pillars: C#’s Encapsulation Services, Pseudo-Encapsulation: Creating Read-Only Fields, The Second Pillar: C#’s Inheritance Supports, keeping Family Secrets: The “Protected” Keyword, Nested Type Definitions, The Third Pillar: C #’s Polymorphic Support, Casting Between.</p> <p>EXCEPTIONS AND OBJECT LIFETIME: Ode to Errors, Bugs, and Exceptions, The Role of .NET Exception Handling, the System.Exception Base Class, Throwing a Generic Exception, Catching Exception, CLR System – Level Exception (System.System Exception), Custom Application-Level Exception (System.System Exception), Handling Multiple Exception, The Family Block, the Last Chance Exception Dynamically Identifying Application – and System Level Exception Debugging System Exception Using VS. NET, Understanding Object Lifetime, the CIT of “new’, The Basics of Garbage Collection,, Finalization a Type, The Finalization Process, Building an Ad Hoc Destruction Method, Garbage Collection Optimizations, The System. GC Type.</p>		

Unit-IV		13 Hours
<p>INTERFACES AND COLLECTIONS: Defining Interfaces Using C# Invoking Interface Members at the object Level, Exercising the Shapes Hierarchy, Understanding Explicit Interface Implementation, Interfaces As Polymorphic Agents, Building Interface Hierarchies, Implementing, Implementation, Interfaces Using VS .NET, understanding the IConvertible Interface, Building a Custom Enumerator (IEnumerable and Enumerator), Building Cloneable objects (ICloneable), Building Comparable Objects (I Comparable), Exploring the system. Collections Namespace, Building a Custom Container (Retrofitting the Cars Type).</p> <p>CALLBACK INTERFACES, DELEGATES, AND EVENTS: Understanding Callback Interfaces, Understanding the .NET Delegate Type, Members of System. Multicast Delegate, The Simplest Possible Delegate Example, , Building More a Elaborate Delegate Example, Understanding Asynchronous Delegates, Understanding (and Using)Events.</p>		
Text Books	:	<ol style="list-style-type: none"> 1) Andrew Troelsen, “Pro C# with .NET 3.0”, Dream Tech Press, 2nd Edition India, 2007. 2) E. Balagurusamy, “Programming in C#, 5th Reprint, Tata McGraw Hill, 2004.
Reference Books	:	<ol style="list-style-type: none"> 1) Tom Archer, “Inside C#”, WP Publishers, 2001. 2) Herbert Schildt, “The Complete Reference C#”, Tata McGraw Hill, 2004

UIS060E: Machine Learning
(4-0-0)(4 Credits, 52 Hours)

- Course Objectives**
1. To introduce students to the basic concepts and techniques of Machine Learning.
 2. To develop skills of using recent machine learning software for solving practical problems.
 3. To gain experience of doing independent study and research.

- Course Outcomes**
- 1 Understand the basic theory underlying machine learning.
 - 2 Understand a range of machine learning algorithms along with their strengths and weaknesses
 - 3 Be able to apply machine learning algorithms to solve problems of moderate complexity
 - 4 Be able to formulate machine learning problems corresponding to different applications.
 - 5 Be able to read current research papers and understand the issues raised by current research

Mapping (Co-relations) of CO's with PO's

Sl. No.	Course Outcomes	IS-PO-1	IS-PO-2	IS-PO-3	IS-PO-4	IS-PO-5	IS-PO-6	IS-PO-7	IS-PO-8	IS-PO-9	IS-PO-10	IS-PO-11	IS-PO-12
01	Understand the basic theory underlying machine learning.	3	3	2		2							2
02	Understand a range of machine learning algorithms along with their strengths and weaknesses	3	3	2		2	2						2
03	Be able to apply machine learning algorithms to solve problems of moderate complexity	2	2	2		2	2						2
04	Be able to formulate machine learning problems corresponding to different applications.	3	2	2		2	2						2
05	Be able to read current research papers and understand the issues raised by current research	3	2	2	2	2	2						2

Syllabus

Academic Year	:	2018 - 19
Semester	:	07
Subject	:	Machine Learning
Subject Code	:	UIS060E
Credits	:	04
Teaching Hours	:	52
Unit –I		13 Hours
Introduction: Basic definitions, types of learning, hypothesis space and inductive bias, evaluation, cross-validation. Linear regression, Decision trees, overfitting		
Unit –II		13 Hours
Instance based learning, Feature reduction, Collaborative filtering based recommendation. Probability and Baye’s learning		
Unit -III		13 Hours
Logistic Regression, Support Vector Machine, Kernel function and Kernel SVM. Neural network: Perceptron, multilayer network, back propagation, introduction to deep neural network		
Unit-IV		13 Hours
Computational learning theory, PAC learning model, Sample complexity, VC Dimension, Ensemble learning. Clustering: k-means, adaptive hierarchical clustering, Gaussian mixture model.		
Text Book		1) Dr. Sudeshna Sarkar, “ Introduction to Machine learning ”, NPTEL Course Material, Department of Computer Science and Engineering, IIT Karagpur
Reference Books		1. Tom Mitchell “ Machine Learning ”, First Edition, McGraw- Hill, May2013. 2. Ethem Alpaydin “ Introduction to Machine Learning ”, PHI publications, 2008

UIS727L: Object Oriented Modeling & Design Laboratory
1.5 CREDIT (0-0-3)

1. Design the Library system: Identify the use cases of the system. (Suggestive use cases: borrow book, return books, read newspapers, reference, and digital library). Develop the use case diagram, Packages and documentation for the same. Preferable use of uses & Extends relationships expected.
2. Design the Examination system: Identify the use cases. (Suggestive use cases:– Form filling, Get Hall Ticket, Write exam, get result Verify Hall Ticket) Develop the use case diagram, Packages and documentation for the same. Preferable use of uses & Extends relationships expected.
3. Analyze and design the system for ATM Transaction: Identify the use cases. (Suggestive use cases: Transaction, Approval process, Invalid PIN, Deposit Amount, Deposit savings, Deposit checking, withdraw Amount, withdraw checking, saving, withdraw saving denied, checking Transaction History, saving Transaction History). Package, documentation Develop the use case diagram, Packages and documentation for the same. Draw the essential class diagrams.
4. Analyze and design the system for Electronics voting system (The actors are presiding officer, 1st polling officer, 2nd polling officer, voters list, Election officer, voter candidate, EVM ID; Processes: Vote counting, and announcement of results). Develop the use case diagram, Packages and documentation for the same. Draw the essential sequence diagrams, activity diagram and state chart diagrams.
5. Analyze and design the system for Employee reference. (The Process HR Manager contacts Employees of his company and HR manager of other company to publicize about the vacancy. The person, who has referred the right candidate, will be given bonus. Interview, Short-listing, selection list announcement, Bonus for referred employees are all parts of the process.). Develop the use case diagram, sequence diagrams and state chart diagrams.
6. Analyze and design the system for Vehicle Purchase, registration and licensing Systems. Develop the use case diagram, sequence diagrams, activity diagrams and packages.
7. Develop State transition diagrams for
 - a. Telephone line System
 - b. Nested State diagram for vehicle transmission states
- 8 Analyze and design the system for autonomous education system. (Classes: students Teacher, courses, subjects, core, Electives, Labs). Develop the class diagrams, sequence diagrams and packages.
9. Analyze and design the system for Results section of autonomy, mainly responsible for CGPA, SGPA Calculation, and Grade card generation. Develop the class diagrams, use case diagram and packages.
10. Analyze and design the system for Railway reservation, which includes the details of Boarding station, classes of reservation, fare, Time Table, concessions, No. of Seats, Seat No, State of reservation (confirmed, waiting, RAC). Develop the class diagrams, packages, and use case diagrams.
11. Analyze and design the system for Income Tax assessment. Develop the class diagrams, use case and state chart diagrams.

General Remarks:

- 1) Lab schedule: 3hrs/week for each student.
- 2) Student should complete all the lab assignments.
- 3) Evaluation CIE 50 marks:
 - Lab assignments : 30 marks
 - Lab CIE : 20 marks

UIS712L: UNIX System Programming Laboratory

1.5 Credits (0-0-3)

Write a program to implement the following:

1. Usage of CPP symbols, POSIX version, POSIX feature test macros, sysconf, pathconf, and fpathconf.
2. Copy operation from input file to output file.
3. Usage of directory file API's to print a directory tree.
4. ls-l command using stat API.
5. Fork and wait system call.
6. SIGCHLD signal, signal API and sigaction API.
7. IPC between sender and receiver process.
8. Socket program.

General Remarks:

- 1) Lab schedule: 3hrs/week for each student.
- 2) Student should complete all the lab assignments.
- 3) Evaluation CIE 50 marks:
Lab assignments : 30 marks
Lab CIE : 20 marks

UIS711P: Project Phase-I

3 Credits

Tasks to be carried out:

1. Identify the team members and the Guide
2. Identify the Problem
3. Literature survey
 - a. Domain knowledge
 - b. Identify the issues/challenges in the domain area
4. Define the problem statement
5. For application oriented projects (non-research projects), students will study and give presentation on technology (tool/language/simulator etc) related to their project work.
6. For research oriented projects,
 - a. Identify the relevant papers (one or two) w.r.t. their problem definition.
 - b. Presentation on the critical analysis of the selected paper(s).
7. Presentation of Pre-project demo.
8. Mid – term demo.
9. Final demo.
10. Preparation of paper on project work (optional).
11. Report preparation and submission.