Kolhapur Institute of Technology's

College of Engineering (AUTONOMOUS), Kolhapur.

(An Autonomous Institute)

Syllabus

T. Y. B. Tech.

Computer Science & Engineering

SEM-I

	L	T	P	Credit
Title of the Course: Computer Algorithm **Course Code:** UCSE0501	**3**	**1**	**-**	**4**

Course Pre-Requisite: Data Structures

Course Description: This course introduces fundamental concepts and key techniques for designing and analyzing algorithms along with study and apply different algorithm design methods namely, greedy method, divide and conquer, dynamic programming and backtracking.

Course Objectives:

1. To introduce to the students the methods of algorithm designs.
2. To expose students to various searching and sorting techniques.
3. To make students understand the analyses of algorithms.
4. To show how to tackle real time problems.

Course Learning Outcomes:

CO	After the completion of the course the student should be able to
CO1	Define basic concepts of algorithms and measure the efficiency of any algorithm.
CO2	Make use of standard design techniques such as divide and conquer, greedy algorithms, dynamic programming, backtracking to solve real life problems.
CO3	Identify graph algorithms to model engineering problems, when appropriate.
CO4	Distinguish between P and NP Classes of problems.

CO-PO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	3	3		2										2
CO2		2	3	3										3
CO3		2	3	3										3
CO4		2		2										

Assessments :

Teacher Assessment:

Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one End Semester Examination (ESE) having 20%, 30% and 50% weights respectively.

Assessment	Marks
ISE 1	10
MSE	30
ISE 2	10
ESE	50

ISE 1 and ISE 2 are based on assignment/declared test/quiz/seminar/Group Discussions etc.

MSE: Assessment is based on 50% of course content (Normally first three modules)

ESE: Assessment is based on 100% course content with60-70% weightage for course content (normally last three modules) covered after MSE.

Course Contents:	
Unit 1: Introduction: What is algorithm, **Algorithm Specification**: Pseudocode Conventions, Recursive Algorithm, **Performance Analysis**: Space Complexity, Time Complexity, Asymptotic Notations, Practical Complexities, Performance Measurement **Recurrences:** The substitution method, recursion tree method	**05 Hrs.**
Unit 2: Algorithm Design and Analysis Techniques – I: **Divide and Conquer**-The general method, Binary search, Finding the maximum and minimum, Merge sort, Quick sort and analysis of these algorithms. **The Greedy method:** The general method, Knapsack problem, Job sequencing with deadlines, Optimal storage on tapes, Optimal merge patterns, Huffman codes.	**08 Hrs.**
Unit 3: Algorithm Design and Analysis Techniques - II: **Dynamic Programming:** The general method, Multistage graphs, Optimal binary search trees, 0/1 knapsack, Reliability design, Travelling Salesperson problem**.**	**07 Hrs.**
Unit 4: Graph Algorithms: **Elementary Graph Algorithms:** Representations of graphs , Breadth-first search, Depth-first search, Strongly connected components, **Minimum Spanning Trees:** Growing a minimum spanning tree, The algorithms of Kruskal and Prim **Single-Source Shortest Paths:** The Bellman-Ford algorithm, Single-source shortest paths in directed acyclic graphs, Dijkstra's algorithm, The Floyd-Warshall algorithm	**07 Hrs.**
Unit 5: Backtracking: Backtracking - The general method, 8-queen problem, Sum of subsets, Graph Coloring, Knapsack Problem, Hamiltonian Cycle.	**07 Hrs.**
Unit 6: NP-Completeness: Polynomial time, Polynomial-time verification, NP-completeness and reducibility, NP-complete problems	**05 Hrs.**
Textbooks: **1.** Thomas Cormen, Charles Leiserson, Ronald Rivest and Cliford Stein, "Introduction to Algorithms", PHI **2.** Fundamentals of Computer Algorithms - Ellis Horowitz, Satraj Sahani, Saguthevar Rajasejaran, Universities Press, Second Edition.	
References: 1. Fundamentals of Algorithmics – Gilles Brassard, Paul Bratley (Pearson Education). 2. Mastering Algorithms with C – Kyle Loudon (SPD O'Reilly). 3. Computer Algorithms- Introduction to Design and Analysis – Sara Baase, Allen Van Gelder (Pearson Education).	
Unit wise Measurable students Learning Outcomes: 1. Students will be able to understand the fundamental concepts in Algorithm design and analysis of an Algorithm. 2. Students will be able to study Divide and Conquer method and analyze the complexity of various algorithms 3. Students will be able to study Greedy method and analyze the complexity of various algorithms. 4. Students will be able to design efficient algorithms for various problems applying Dynamic programming method. 5. Students will be able to study Backtracking method and identify the various problems which can be solved by using Backtracking method 6. Students will be able to design Efficient Algorithm for Tree and Graph traversal techniques. 7. Students will be able to understand the NP-Problems.	

Title of the Course: Operating System **Course Code: UCSE0502**	**L**	**T**	**P**	**Credit**
	3	--	--	3

Course Pre-Requisite: Fundamentals of Electronics and Computer

Course Description: This course is introduced at third year level to get the students familiar with the basic concepts of computer operating systems in detail.

Course Learning Objectives:

clo1. To make the students understand basic concepts of operating system.

clo2. To expose the students to various functions of the Operating system and their usage.

clo3. To make the students understand CPU scheduling, Memory management.

Course Outcomes:

CO	**After the completion of the course the student should be able to**
CO1	Define the features and types of operating system.
CO2	Explain process related concepts such as scheduling, communication and synchronization.
CO3	Analyze memory management techniques for primary as well as secondary memory
CO4	Compare algorithms for CPU scheduling, Disk Scheduling and page replacement policies.

CO-PO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PO 12	PSO1	PSO 2
CO1	2	-	-	-	-	-	-	-	-	-	-	-	1	-
CO2	2	-	-	-	-	-	-	-	-	-	-	-	2	-
CO3	-	2	-	-	-	-	-	-	-	-	-	-	2	-
CO4	-	2	-	2	-	-	-	-	-	-	-	-	2	-

Assessments :

Teacher Assessment:

Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one EndSemester Examination (ESE) having 20%, 30% and 50% weights respectively.

Assessment	Marks
ISE 1	10
MSE	30
ISE 2	10
ESE	50

ISE 1 and ISE 2 are based on assignment/declared test/quiz/seminar/Group Discussions etc.

MSE: Assessment is based on 50% of course content (Normally first three modules)

ESE: Assessment is based on 100% course content with60-70% weightage for course content (normally last three modules) covered after MSE.

Course Contents:

Unit 1:--- Introduction to Operating Systems Abstract view of an operating system, Fundamental principles of OS operations, OS interaction with the computer and user programs, Efficiency ,system performance and user service, Batch Processing System, Multiprogramming System, The Time Sharing System, The Real Time Operating System, Operation of OS, Operating system with monolithic structure, Virtual machine operating system, Kernel based operating system, Microkernel based operating system	**6 Hrs.**
Unit 2:--- Process Management Process Concept; Process Scheduling; Operations on Processes; Inter-Process Communication; Multi-Threaded Programming Overview; Multithreading Models **Process Scheduling**: Basic Concepts; Scheduling Criteria; Scheduling Algorithms; Multiple-Processor Scheduling; Thread Scheduling	**6 Hrs.**
Unit 3:--- Process Synchronization Synchronization: The Critical Section Problem; Peterson's Solution; Synchronization Hardware; Semaphores; Monitors, Classical Problems of Synchronization, Monitors	**8 Hrs.**

Unit 4:--- Deadlocks System Model; Deadlock Characterization; Methods for Handling Deadlocks; Deadlock Prevention; Deadlock Avoidance; Deadlock Detection and Recovery from Deadlock	**7 Hrs.**
Unit 5:--- Memory Management Memory Management Strategies: Background; Swapping; Contiguous Memory Allocation; Paging; Structure of Page Table; Segmentation; Virtual Memory Management: Background; Demand Paging; Copy-on-Write; Page Replacement; Allocation of Frames; Thrashing	**7 Hrs.**
Unit 6:--- Disk Management and IO Management Disk Structure, Disk Scheduling Algorithms Overview, I/O Hardware, Application I/O Interface, Kernel IO Subsystem, Transforming I/O Request to Hardware Operations	**6 Hrs.**

Textbooks:

1. Operating Systems A Concept Based Approach 3rd Edition, Mc-Graw Hill, Unit No.1

2. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne: Operating System Principles, 8th edition, Wiley India, 2009

References:

1. Operating Systems –Concepts and design –Milan Milenkovic (TMGH) (For Types of Operating Systems - Refer Chapter 1 in Operating Systems – Concepts and design – Milan Milenkovic (TMGH))

2. Operating Systems: Internals and Design Principles (8th Edition)- by William Stallings(Pearson Education International)

3. Modern Operating Systems by Andrew S. Tanenbaum (Pearson Education International)

Unit wise Measurable students Learning Outcomes:

1: Introduction to Operating Systems

After successful completion of these unit students will be able to:

UO-1: Explain basic concepts of operating systems
UO-2: Identify different types of operating systems
UO-3: Describe various operating system services
UO-4: Elaborate operating system structure

2: Process Management

After successful completion of these unit students will be able to:

UO-1: Illustrate basic concepts about processes
UO-2: Explain inter-process communication
UO-3: Define multithreaded programming concepts
UO-4: Elaborate process scheduling

3: Process Synchronization

After successful completion of these unit students will be able to:

UO-1: Elaborate the basic concepts of process synchronization
UO-2: State and explain the critical section problem with its solution
UO-3: Describe classical problems of synchronization
UO-4: Distinguish different synchronization hardware

Unit IV: Deadlocks

After successful completion of these unit students will be able to:

UO-1: Describe the basic concepts of deadlocks
UO-2: Illustrate the mechanism to handle deadlocks
UO-3: Examine the prevention and avoidance of deadlocks
UO-4: Explain the detection and recovery of deadlocks

Unit V: Memory Management

After successful completion of these unit students will be able to:

UO-1: State and explain various memory management strategies
UO-2: Describe the concepts like paging and segmentation
UO-3: Elaborate the virtual memory management schemes
UO-4: Illustrate various page replacement policies

Unit VI: Disk management and IO Management

After successful completion of these unit students will be able to:

UO-1: Elaborate the disk scheduling policies
UO-2: Describe various IO hardware components
UO-3: Examine kernel IO subsystems
UO-4: Describe transformation of IO request to hardware operation

Title of the Course: Mobile Technology **Course Code: UCSE0521**	**L**	**T**	**P**	**Credit**
	3	**-**	**-**	**3**

Course Pre-Requisite: Data Communication & Networking
TCP/IP Protocol Suite

Course Description: Mobile Technology subject mainly deals with the science of mobile communication. It covers layered approach of mobile communication covering layers such as – data link, network and transport layer. It also introduces the 5G technology.

Course Learning Objectives:

To expose students to:

1. Mobile Computing basics & GSM architecture
2. Network Layer & Transport Layer of Mobile communication
3. Mobile Adhoc network basics and routing.
4. Fundamentals of the 5G Mobile Network

Course Outcomes:

CO	**After the completion of the course the student should be able to**
CO1	Explain the basic physical and technical functioning of mobile communications systems
CO2	Describe the basic principles of mobile communication system and summarize the working of network and transport layer in the context of mobility
CO3	Compare principles of the modern mobile and wireless communication systems such as 5G with 3G/4G
CO4	Illustrate the principles and applications of wireless systems and standards

CO-PO Mapping:

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
CO1	2													
CO2	1	2												2
CO3	1	2		2	2									
CO4	1	1												

Assessments :

Teacher Assessment:

Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one End Semester Examination (ESE) having 20%, 30% and 50% weights respectively.

Assessment	Marks
ISE 1	10
MSE	30
ISE 2	10
ESE	50

ISE 1 and ISE 2 are based on assignment/declared test/quiz/seminar/Group Discussions etc.
MSE: Assessment is based on 50% of course content (Normally first three modules)
ESE: Assessment is based on 100% course content with 60-70% weight age for course content (normally last three modules) covered after MSE.

Course Contents:

Unit 1: Introduction Introduction to Wireless Networks, Applications, History, Simplified Reference Model, Wireless transmission, Frequencies, Signals, Antennas, Signal propagation, Multiplexing, Modulation, Spread spectrum, Cellular Systems: Frequency Management and Channel Assignment, Types of handoff and their characteristics.	**06 Hrs.**
Unit 2: Medium Access Control (MAC) & GSM Telecommunication System MAC, Motivation, SDMA, FDMA, TDMA, CDMA, Telecommunication Systems, GSM: Architecture, Location tracking and call setup, Mobility management, Handover, Security, GSM SMS, International roaming for GSM, call recording functions, subscriber and service data management.	**08 Hrs.**
Unit 3: Mobile Network Layer: IP and Mobile IP Network Layers, Packet Delivery and Handover Management, Location Management, Registration, Tunneling and Encapsulation, Route Optimization, DHCP.	**06 Hrs.**
Unit 4: Mobile Transport Layer Traditional TCP, Indirect TCP, Snooping TCP, Mobile TCP, Fast retransmit / Fast recovery, Transmission / Timeout freezing, Selective retransmission, Transaction Oriented TCP.	**06 Hrs.**
Unit 5: Mobile Ad hoc Networks (MANETs) : Introduction, Applications & Challenges of a MANET, Routing, Classification of Routing Algorithms, Algorithms such as DSR, AODV, DSDV, etc. , Mobile Agents, Service Discovery.	**06 Hrs.**
Unit 6: 5G Mobile Network – Future of Mobile Communication 10 Pillars of 5G, concept of small cell, Cognitive radio -overview, spectrum optimization literature, key requirements and challenges for 5G cognitive terminal. Wireless spectrum white spaces – Background, TV white space technology, white space spectrum opportunities and challenges.	**06 Hrs.**
Textbooks: 1. Jochen Schiller, \Mobile Communication", Pearson Education. 2. Theodore & S. Rappaport, \Wireless Communications, Principles, Practice", PHI. 3. William Stallings, \Wireless Communications and Networks", Pearson Education. 4.Jonathan Rodriguez, Fundamentals of 5G Mobile Networks, First Edition 2015 John Wiley & Sons, Ltd Publication	
References: 1. Wireless telecommunications systems and networks / Gary J. Mullett. Cengage Publication.	

Title of the Course: Programming Paradigm	**L**	**T**	**P**	**Credit**
Course Code:UCSE0522	3	--	--	3

Course Pre-Requisite: Introductory information in Computer Architecture.

Course Description: This subject is introduced to teach the major programming paradigms, and the principles and techniques involved in design and implementation of modern programming languages.

Course Learning Objectives:

1. To briefly describe various programming paradigms.
2. To provide conceptual understanding of High level language design and implementation.
3. To introduce the power of scripting languages

Course Outcomes:

CO	**After the completion of the course the student should be able to**
CO1	Show syntax and semantics in formal notation.
CO2	Apply suitable programming paradigm for the application.
CO3	Analyze the features of different programming languages

CO-PO Mapping:

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**	**PSO1**	**PSO2**
CO1	**2**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**
CO2	**-**	**-**	**2**	**-**	**3**	**-**	**-**	**1**	**-**	**-**	**-**	**-**	**-**	**1**
CO3	**1**	**2**	**-**	**-**	**2**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**

Assessments :

Teacher Assessment:

Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one End Semester Examination (ESE) having 20%, 30% and 50% weights respectively.

Assessment	Marks
ISE 1	10
MSE	30
ISE 2	10
ESE	50

ISE 1 and ISE 2 are based on assignment/declared test/quiz/seminar/Group Discussions etc.

MSE: Assessment is based on 50% of course content (Normally first three modules)

ESE: Assessment is based on 100% course content with60-70% weightage for course content (normally last three modules) covered after MSE.

Course Contents:

Unit 1: Introduction Introduction to Programming paradiagms, Bridging the gap. Language Description: Expression Notations, Abstract syntax trees, Lexical Syntax, Context-free Grammers,Grammars for expressions	6 **Hrs**
Unit 2: Imperative Programming The need for Structured Programming, Syntax-Directed control flow, Design Considerations: syntax, Handling special cases in loops, Control flow in C Data Representation: The role of types, Basic types, Arrays, Records, Unions & Variant Records,Sets, Pointers.	6 **Hrs**
Unit 3: Procedure Programming Introduction to Procedures, Prameter-passing methods, Scope rules for names,Nested Scopes in the source text, Activation Records, Lexical scope: Procedures as in C	6 **Hrs.**
Unit 4: Object-oriented programming Constructs for program structuring, Information Hiding, Program design with modules, Modules & Defined types, Class declartions in C++, dynamic allocation in C++,what is an object? Object oriented thinking, Inheritance, Object oriented programming in C++, example: C++, Java.	7 **Hrs.**
Unit 5: Functional programming A little language of expressions, Types: Values and operations, Approaches to expression Evaluation, Lexical scope, Type checking, Function declaration by cases, Functions as First class values, ML: Implicit types, Data types.	7 **Hrs.**
Unit 6: Concurrent and logical Programming Logic Programming: Introduction to Prolog, Data Structures in Prolog, Programming techniques. Concurrent Programming: Parallelism in hardware, Streams: Implicit synchronization, Concurrency as Interleaving, Liveness Properties, Concurrency in ADA, Exception Handling, Example: Parallel Java, Parallel C.	6 **Hrs.**

Textbooks:

1. Programming Languages: Concepts & Constructs, 2/E, Ravi Sethi
2. Programming Languages: Principles and Paradigms by Maurizio Gabbrielli and Simone MarPni

References:

1. Programming Language Pragma@cs by Michael L. Sco_, 3rd edition
2. Programming Language Pragmatics. Michael Scott, Morgan Kaufmann, 2000

Unit wise Measurable students Learning Outcomes:

1. Students should be able to learn basic language descriptions
2. Students should be able to learn imperative programming structure
3. Students should be able to learn concepts of procedural programming
4. Students should be able to learn structure of Object-oriented programming
5. Students should be able to learn concepts of functional programming
6. Students should be able to learn concepts of Concurrent & Logical programming

Title of the Course: Project Management Course Code: UCSE0523	L	T	P	Credit
	3	-	-	3

Course Pre-Requisite: Software Engineering

Course Description:
This course develops a foundation of concepts and solutions that supports the planning, scheduling, controlling, resource allocation, and performance measurement activities required for successful completion of a project.

Course Objectives:

1. To provide students with a basic understanding of project management principles and practices.
2. To demonstrate competency in the creation and management of a project plan
3. To understanding impact of Scope, Time and Cost management.
4. To develop strategies to calculate risk factors involved in IT projects.
5. To Use tools and techniques for project management.

Course Learning Outcomes:

CO	After the completion of the course the student should be able to
CO1	Explain basic concept of project management.
CO2	Make use of tools and techniques for project activities.
CO3	Inspect reason for project failures.
CO4	Design project management plan for real world problem.

CO-PO Mapping:

CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PS01	PSO2
CO1											2			
CO2					3								2	1
CO3		3											2	1
CO4			3						2				3	1

Assessments :
Teacher Assessment:
Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one End Semester Examination (ESE) having 20%, 30% and 50% weights respectively.

Assessment	Marks
ISE 1	10
MSE	30
ISE 2	10
ESE	50

ISE 1 and ISE 2 are based on assignment/declared test/quiz/seminar/Group Discussions etc.
MSE: Assessment is based on 50% of course content (Normally first three modules)
ESE: Assessment is based on 100% course content with60-70% weightage for course content (normally last three modules) covered after MSE.

Course Contents:	
Unit 1: Introduction to Project Management: (6) Project and Project Management (PM), Role of project Manager, System view of PM,Organization, Stakeholders, Project phases and lifecycle, Context of IT projects, process groups, mapping groups to Knowledge areas	**6 Hrs.**
Unit 2:- Project Integration and Scope Management: (8) Strategic planning and project selection, Developing a Project Management Plan, Directing and Managing Project Work, Monitoring and Controlling Project Work, Performing Integrated Change Control, Closing Projects or Phases Planning Scope Management, Collecting Requirements, Defining Scope, Creating the Work Breakdown Structure, Validating Scope, Controlling Scope	**8 Hrs.**
Unit 3:-Project Time Management: (4) Planning Schedule Management, Defining Activities, Sequencing and Estimating Activity, Resources & Duration, Developing & Controlling Schedule	**4 Hrs.**
Unit 4:- Project Cost and Risk management: (6) Basic Principles of Cost Management, Planning Cost Management, Estimating Costs, Determining the Budget, Controlling Costs Importance, risk management planning, sources of risk, risk identification, qualitative and quantitative risk analysis, risk response planning, risk monitoring and control.	**6 Hrs.**
Unit 5:- Project Procurement Management (7) The Importance of Project Procurement Management , Planning Procurements , Tools and Techniques for Planning Procurements , Procurement Management Plan ,Statement of Work , Procurement Documents , Source Selection Criteria , Conducting Procurements Administering Procurements , Closing Procurements , Using Software to Assist in Project Procurement Management	**7Hrs.**
Unit 6:- Using OpenProject software for project management**(6)** Introduction and Overview of New Features of OpenProject software, getting started with OpenProject 2010 ,Using the Help Feature ,Main Screen Elements , Project 2010 Views , Project 2010 Filters, Developing a Work Breakdown Structure, Gantt Charts ,Network Diagrams ,Critical Path Analysis.	**6 Hrs.**
Textbooks:1. Information Technology Project Management, 7E, Kathy Schwalbe, Cengage Learning 2. https://opensource.com/article/17/11/how-install-and-use-openproject	
References: 1. The principles of project management by MERI WILLIAMS	
Unit Wise Measurable students learning Outcomes: 1. Student will be able to Manage the scope, cost, timing, and quality of the project, at all times focused on project success as defined by project stakeholders. 2. Student will be able to Identify project goals, constraints, deliverables, performance criteria, control needs, and resource requirements in consultation with stakeholders. 3. Student will be able to Implement project management knowledge, processes, lifecycle and the embodied concepts, tools and techniques in order to achieve project success. 4. Student will be able to Adapt project management practices to meet the needs of stakeholders from multiple sectors of the economy (i.e. consulting, government, arts, media, and charity organizations). 5. Student will be able to Apply project management practices to the launch of new programs, initiatives, products, services, and events relative to the needs of stakeholders.	

Title of the Course: Database Engineering **Course Code:** UCSE0503	L	T	P	Credit
	3			3

Course Pre-Requisite: Data Structures

Course Description: This course introduces database design and creation using a DBMS product. Emphasis is on data dictionaries, normalization, data integrity, data modelling, and creation of simple tables and queries. Upon completion, students should be able to design and implement normalized database structures by creating simple database tables and queries.

Course Objectives:

1. To understand Fundamental Concepts related to database.
2. To gain familiarity with SQL & DBMS.
3. To understand basic concepts of Database Design

Course Learning Outcomes:

CO	After the completion of the course the student should be able to
CO1	Define basic functions and features of DBMS & RDBMS.
CO2	Apply normalization techniques on given database to assure data integrity and consistency.
CO3	Make use of SQL queries to implement user defined applications.
CO4	Interpret file organization, indexing and hashing techniques for faster and efficient system performance.
CO5	Distinguish transaction management and concurrency control methods for system reliability and security.

CO-PO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	2													
CO2		2	2										2	3
CO3		2	3	3	3								2	3
CO4	3			2									2	3
CO5	2	2											3	3

Assessments :

Teacher Assessment:

Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one End Semester Examination (ESE) having 20%, 30% and 50% weights respectively.

Assessment	Marks
ISE 1	10
MSE	30
ISE 2	10
ESE	50

ISE 1 and ISE 2 are based on assignment/declared test/quiz/seminar/Group Discussions etc.

MSE: Assessment is based on 50% of course content (Normally first three modules)

ESE: Assessment is based on 100% course content with60-70% weightage for course content (normally last three modules) covered after MSE.

Course Contents:

Unit 1: Introduction: Purpose of Database Systems, View of Data, Data Models, Database Architecture, Roles in Database Environment, The Entity-Relationship Model,	**08 Hrs.**

Entity-Relationship Diagrams, Reduction to Relational Schemas, Introduction to Relational Model, Relational Query Languages- The Relational Algebra,	
Unit 2: Relational Database Design: The purposes of Normalization, Data Redundancies and Update Anomalies, Functional Dependencies, The Process of Normalization, First Normal Form, Second Normal Form, Third Normal Form, Boyce-Codd Normal Form, Fourth Normal Form, Fifth Normal Form.	**06 Hrs.**
Unit 3: Relational Model and Structured query Language Structure of Relational Databases, SQL Data Definition Language, Basic Structure of SQL Queries, Additional Basic Operations, Set Operations, Null Values, Aggregate Functions, Nested Subqueries, Modification of the Database, Join Expressions, Views, Integrity Constraints, Accessing SQL from a Programming Language.	**06 Hrs.**
Unit 4: File Structure, Indexing and Hashing: Overview of Physical Storage Media, File Organization, Organization of Records in Files, Data-Dictionary Storage, Database Buffer. Basic Concepts of Indexing and Hashing, Ordered Indices, B+-Tree Index Files, B-Tree Index Files, Multiple-Key Access, Static Hashing, Dynamic Hashing, Bitmap Indices, Index Definition in SQL.	**08 Hrs.**
Unit 5: Transactions and Concurrency Control: Transaction Concept, Simple Transaction Model, Serializability, Concurrency Control- Lock-Based Protocols, Two-phase locking protocols, Graph-based protocols, Multiple Granularity, Timestamp-Based Protocols, Validation-Based Protocols.	**06 Hrs.**
Unit 6: Recovery System : Failure Classification, Storage, Recovery and Atomicity, Recovery Algorithm, checkpoint, Shadow paging, Failure with Loss of Non-volatile Storage, Remote Backup Systems	**06 Hrs.**
Textbooks: 1. DataBase System Concept by Henry F. Korth, Abraham Silberschatz, Sudarshan (McGraw Hill Inc.) Sixth Edition. 2. Database Systems- A practical approach to Design, Implementation and Management by Thomos Connolly, Carolyn Begg, 3rd Edition, Pearson Education	
References: 1. Fundamentals of Database Systems – by Ramez Elmasri and Shamkant Navathe Publisher -Pearson Education, 5 th Edition. 2. Database Systems: Design, Implementation and management.- PeterRof, Carlos Coronel (7th Edition), Publisher - Cengage Learning. 3. Principles of Database Systems by J.D. Ullaman (Galgotia Publications).	
Unit wise Measurable students Learning Outcomes: 1. Student will be able to understand fundamental concepts of DBMS and Data models. 2. Student will be able to write SQL query to perform different database applications. 3. Student will be able to apply normalization techniques to create good database design. 4. Student will be able to understand concept of file organization, indexing and hashing. 5. Student will be able to understand concurrency control management. 6. Student will be able to apply recovery algorithm for reliability of database.	

Title of the Course: Machine Learning	**L**	**T**	**P**	**Credit**
Course Code: UCSE0504	2	1	--	3

Course Pre-Requisite: Discrete Mathematics, Mathematics for Computer Science

Course Description:

Course Learning Objectives:

1. To understand Human learning aspects.
2. To understand primitives in learning process by computer.
3. To understand nature of problems solved with Machine Learning

Course Outcomes:

CO	**After the completion of the course the student should be able to**
CO1	Explain machine learning concepts.
CO2	Analyse the Machine learning model.
CO3	Design solution using machine learning techniques

CO-PO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1	1	1												
CO2		2												
CO3		2	2		1								1	1

Assessments :

Teacher Assessment:

Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one EndSemester Examination (ESE) having 20%, 30% and 50% weights respectively.

Assessment	Marks
ISE 1	10
MSE	30
ISE 2	10
ESE	50

ISE 1 and ISE 2 are based on assignment/declared test/quiz/seminar/Group Discussions etc.

MSE: Assessment is based on 50% of course content (Normally first three modules)

ESE: Assessment is based on 100% course content with60-70% weightage for course content (normally last three modules) covered after MSE.

Course Contents:

Unit 1:- Introduction. Introduction to Machine Learning: Definition, Terminology, Types of learning, Machine Learning Problem categories, Machine learning architecture, process, Lifecyle, Performance measures, tools and framework, data visualization	**6 Hrs.**
Unit 2:- Regression: Simple regression – hypothesis, cost function, parameter learning with gradient descent, learning rate, Gradient Descent for linear regression, examples, simple regression in matrix form. Multivariate Linear Regression – multiple features, hypothesis functions, Gradient Descent for multiple variables, Feature scaling, polynomial regression	**6 Hrs.**
Unit 3: Classification- logistic regression & Neural Network: Definition, logistic regression – hypothesis representation, decision boundary, cost	**4 Hrs.**

function, gradient descent for logistic regression. multiclass classification, Regularization - Overfitting & Underfitting, cost function, Regularized Linear Regression, Regularized Logistic Regression Neural Networks- Neuron representation and model, Hypothesis for neuron, cost function, solution of a problem using single neuron. Gradient descent for a neuron. Neural network, Multiclass classification with neural network. Learning in neural network-backpropagation algorithm	
Unit 4: Classification- Decision trees and Naïve Bayes Decision trees: definition, terminology, the need, advantages, and limitations. constructing and understanding Decision trees, common problems with Decision trees, Decision tree algorithms, random forest, examples. Conditional probability and Naïve Bayes Classifier Instance-based classifier – K- Nearest Neighbour Classifier	**7 Hrs.**
Unit 5 Unsupervised learning: Clustering, K Means clustering, Hierarchical clustering, Association Rule mining,	**5 Hrs.**
Unit 6: Introduction to Graphical Models Bayesian Networks, Markov Random Field (MRF), Hidden Markov Model, Training HMM: Viterbi, Baum-welch algorithm	**7 Hrs.**
Textbooks: 1. Machine Learning with Python- an approach to applied ML, by Abhishek Vijayvargia, BPB publications 2. Practical Machine Learning by Sunila Gollapudi Packt Publishing Ltd 3. Machine Learning by Tom M. Mitchell, McGraw Hill Education; First edition	
References: 1. http://neuralnetworksanddeeplearning.com/. 2. Machine Learning for dummies John Paul Muller, Willey Publication 3. EthemAlpaydin : Introduction to Machine Learning, PHI 2nd Edition-2013	
Unit wise Measurable students Learning Outcomes:	

	L	T	P	Credit
Title of the Course: Database Engineering Lab	**L**	**T**	**P**	**Credit**
Course Code: UCSE0531			**2**	**1**

Course Pre-Requisite: Data Structures, Programming Language

Course Description: This course is designed to develop SQL programming expertise. Upon completion, students should be able to write programs for database connectivity. Emphasis is on data definition, data manipulation, and data control statements.

Course Objectives:

1. To understand Fundamental Concepts related to database.
2. To gain familiarity with SQL & DBMS.
3. To understand basic concepts of Database Design.

Course Learning Outcomes:

CO	After the completion of the course the student should be able to
CO1	Design conceptual models of a database using ER modelling for real life applications and also construct queries in Relational Algebra
CO2	Apply Normalization to generate good database design
CO3	Develop a database for any specified domain according to well-known design principles.

CO-PO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1		2	3		3			2	3	3	3	2	2	2
CO2	2	2						2	3	3	3	2		2
CO3		2	2	3	3			2	3	3	3	2	3	2

Assessments :

Teacher Assessment:

Two components of In Semester Evaluation (ISE) and one End Semester Examination (ESE) having 33%, and 67% weights respectively.

Assessment	Marks
ISE	25
ESE	50

ISE are based on practical performed/ Quiz/ Mini-Project assigned/ Presentation/ Group Discussion/ Internal oral etc.

ESE: Assessment is based on practical-oral examination.

Course Contents:

Experiment No. 1: Entity - Relationship Diagrams **Aim and Objectives:** Draw ER diagrams for different organizations using any suitable software & Convert them into tables. **Theoretical Background:** Study of Entity-Relationship Diagrams, Reduction to Relational Schemas **Experimentation:** 1. Install Dia software 2. Use E-R sheet to draw E-R diagram.	**02 Hrs.**

3. Convert each E-R diagram to relational schema.	
Experiment No. 2: Relational Algebra **Aim and Objectives:** Solve different queries with relational algebra. **Theoretical Background:** Relational Query Languages- The Relational Algebra. **Experimentation:** 1. Write relational algebra queries with its operations.	**02Hrs.**
Experiment No. 3: Functional Dependencies **Aim and Objectives:** Write program to find closure for a given set of functional dependencies and closure of attribute set. **Theoretical Background:** Normalization and Functional Dependencies. **Experimentation:** 1. Consider a set of functional dependency as input 2. Apply armstrong's axioms and find closure of Functional Dependencies. 3. Find attributes closure.	
Experiment No.4: Installation of Database software (PostgeSQL/ MySQL/Oracle/SQL Server - any one of these) **Aim and Objectives:** Installing Database Software, Administrating it and Creating Users, Connecting to Database Software. **Theoretical Background:** Structured Query Language **Experimentation:** 1. Installing Database Software 2. Create Database 3. Create user with password 4. create schema	**02Hrs.**
Experiment No. 5: Data Definition Language **Aim and Objectives:** Use DDL Queries to create, alter and drop tables with respect to all types constraints(key, referential, not null) **Theoretical Background:** Data Definition Language **Experimentation:** 1. Execute DDL command to create, alter and drop tables in SQL, 2. Apply all types of constraints such as primary key, foreign key, not null, etc.	**02Hrs.**
Experiment No. 6: Data Manipulation Language **Aim and Objectives:** Use DML Queries to insert, delete and update records of the tables. **Theoretical Background:** Modification of the Database **Experimentation:** 1. Execute DML command on the table created in experiment no.5	**02Hrs.**
Experiment No. 7: SQL Query Processing **Aim and Objectives:** Display the records using group by, order by, having and between clauses. **Theoretical Background:** Basic Structure of SQL Queries, Groupby, Orderby clause. **Experimentation:** 1. Write Java program in eclipse for Database Connectivity 2. Execute SQL queries through java eclipse on the table created in experiment no.5	**02Hrs.**
Experiment No. 8: SQL Query Processing **Aim and Objectives:** Display the results of union, intersection, set difference, Cartesian product and Join operations. **Theoretical Background:** SQL set operations and join operations **Experimentation:** 1. Write Java program in eclipse for Database Connectivity 2. Execute SQL queries through java eclipse on the table created in experiment no.5	**02Hrs.**
Experiment No. 9: SQL Query Processing **Aim and Objectives:** Display the records using Aggregate functions and Create Indexes & Views for the table. **Theoretical Background:** SQL aggregate functions, index and views. **Experimentation:** 1. Write Java program in eclipse for Database Connectivity 2. Execute SQL queries through java eclipse on the table created in experiment no.5	**02Hrs.**
Experiment No. 10: Static Hashing **Aim and Objectives:** Write a program to implement Static Hashing.	**02 Hrs.**

Theoretical Background: Indexing and Hashing **Experimentation:** 1. Consider any one table as input created in experiment no.5 2. Select search key 3. Apply hash function 4. Find hash value and put record in appropriate bucket.	
Experiment No. 11: Concurrency Control **Aim and Objectives:** Write a program to simulate any one concurrency control protocol. **Theoretical Background:** Concurrency Control- Lock-Based Protocols **Experimentation: 1.** Consider any one table as input created in experiment no.5 2. Create two programs one for shared lock and another for exclusive lock 3. Show result of compatibility matrix.	**02 Hrs.**
Experiment No. 12: Database Logs **Aim and Objectives:** Write program to create logs of the different activities. **Theoretical Background:** Recovery and Atomicity **Experimentation:** 1. Consider any one transaction with basic operation 2. Create deferred and immediate logs.	**02 Hrs.**
Textbooks: 1. DataBase System Concept by Henry F. Korth, Abraham Silberschatz, Sudarshan (McGraw Hill Inc.) Sixth Edition. 2. Database Systems- A practical approach to Design, Implementation and Management by Thomos Connolly, Carolyn Begg, 3rd Edition, Pearson Education	
References: 1. Fundamentals of Database Systems – by Ramez Elmasri and Shamkant Navathe Publisher -Pearson Education, 5th Edition. 2. Database Systems : Design, Implementation and management.- PeterRof, Carlos Coronel (7th Edition), Publisher - Cengage Learning. 3. Principles of Database Systems by J.D. Ullaman (Galgotia Publications)	
Experiment wise Measurable students Learning Outcomes: 1. Students will be able to use Dia software to create E-R diagram 2. Students will be able to install Database Software. 3. Students will be able to apply normalization techniques for generation of good database design. 4. Student will be able to write SQL queries for data definition, data manipulation, and data control statements. 5. Students will be able to write program for database connectivity using java eclipse as front end and SQL/Oracle as backend.	

Title of the Course: Advanced Programming Laboratory Course Code: UCSE0532	L	T	P	Credit
	2		4	3

Course Pre-Requisite:

1. Object Oriented Programming using C++ Lab (Object Oriented Programming concepts)
2. Basic Networking concepts
3. Threading concepts

Course Description : In this course students will be introduced to strict oop programming environment of Java programming language. Students will learn advanced feature of Java, such as platform independent architecture, JVM, JIT components. The course will also enable students to develop GUI based computer applications which will make use of advanced computer features, for example multi threaded application, networking based applications, database application.

Course Objectives:

1. To explain fundamental and object oriented concepts of Java.
2. To distinguish OOP concepts implementation in Java compared to C++.
3. To expose students to advanced features in Java.
4. To develop GUI applications using Java such as chatting server, student inventory management etc.

Course Learning Outcomes:

CO	After the completion of the course the student should be able to
CO1	Use knowledge of fundamental and oop concepts for programming.
CO2	Apply knowledge of various concepts of computer science and design solutions for different subjects like computer algorithm, threading, networking, database.
CO3	Develop simple applications. Example. Developing application to maintain students basic profile.

CO-PO Mapping:

CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO2
CO 1	3												1	1
CO 2	2		2		2			1				1	1	1
CO 3			3		3			1				1	2	3

Assessments :

Teacher Assessment:

One component of In Semester Evaluation (ISE) and one End Semester Examination (ESE) having 50%, and 50% weights respectively.

Assessment	Marks
ISE	25
ESE	50

ISE: Assignments (15 Marks), Oral (5 Marks), Quiz (5 Marks)

ESE: Assessment is based on practical and oral examination

Course Contents:

Unit 1:--- Fundamental Programming in Java : The Java Buzzwords, The Java Programming Environment- JVM, JIT Compiler, Byte Code Concept, HotSpot, A Simple Java Program, Source File Declaration Rules, Comments, Data Types, Variables, Operators, Strings, Input and Output, Control Flow, Big Numbers, ArraysJagged Array. Objects and Classes: Object-Oriented Programming Concepts, Declaring Classes, Declaring Member Variables, Defining Methods, Constructor, Passing Information to a Method or a Constructor, Creating and using objects, Controlling Access to Class Members, Static Fields and Methods, this keyword, Object Cloning, Class Design Hints.	**4 Hrs.**
Unit 2:--- Interface, Inheritance and Packaging : Interfaces: Defining an Interface, Implementing an Interface, Using an Interface as a Type, Evolving Interfaces, Default Methods. Inheritance: Definition, Superclasses, and Subclasses, Overriding and Hiding Methods, Polymorphism, Inheritance Hierarchies, Super keyword, Final Classes and Methods, Abstract Classes and Methods, casting, Design Hints for Inheritance, Nested classes & Inner Classes, finalization and garbage collection. Packages: Class importing, Creating a Package, Naming a Package, Using Package Members, Managing Source and Class Files. Developing and deploying (executable) Jar File.	**4 Hrs.**
Unit 3:--- Exception and I/O Streams: Exception: Definition, Dealing with Errors, The Classification of Exceptions, Declaring Checked Exceptions, Throw an Exception, Creating Exception Classes, Catching Exceptions, Catching Multiple Exceptions, Re-throwing and Chaining Exceptions, finally clause, Advantages of Exceptions, Tips for Using Exceptions. I/O Streams: Byte Stream – InputStream, OutputStream, DataInputStream, DataOutputStream, FileInputStream, FileOutputStream, Character Streams, BufferedStream, Scanner, File, RandomAccesFile.	4 **Hrs.**
Unit 4:-- Graphical User Interfaces using Swing: Introduction to the Swing, Swing features, Swing Top Level Containers-Creating a Frame, Positioning a Frame, Displaying Information in a Panel, The Model-View-Controller Design Pattern, The JComponent Class. Layout Management: Introduction to Layout Management, APIs for Border Layout, Flow Layout, Grid Layout Event Handling: Basics of Event Handling, The AWT Event Hierarchy, Semantic and Low-Level Events in the AWT, Low-Level Event Types User Interface Components: Text Input, Choice Components, Menus, Dialog Boxes Setting the Look and Feel of UI, Introduction to JApplet	6 **Hrs.**
Unit 5:--- Networking and Multithreading: Networking: Overview of Networking, Networking Basics, Working with URLs, Creating a URL, Parsing a URL, Reading Directly from a URL, Connecting to a URL, Reading from and Writing to a URL Connection, Sockets, Reading from and Writing to a Socket, Writing the Server Side of a Socket, Datagrams, Writing a Datagram Client and Server. Multithreading: Processes and Threads, Runnable Interface and Thread Class , Thread Objects, Defining and Starting a Thread, Pausing Execution with Sleep, Interrupts, Thread States, Thread Properties, Joins, Synchronization	**3 Hrs.**
Unit 6:--- Collection and Database Programming: Collections: Collection Interfaces, Concrete Collections- List, Queue, Set, Map, the Collections Framework. Database Programming: The Design of JDBC, The Structured Query	4 **Hrs.**

Language, JDBC Installation, Basic JDBC Programming Concepts, Query Execution, Scrollable and Updatable Result Sets, Metadata, Row Sets, Transactions

Textbooks:

1. Core Java- Volume I Fundamentals: Cay Horstmann and Gary Cornell, Pearson, Eight edition (Unit 1 to Unit 4).
2. Core Java- Volume II Advanced Features: Cay Horstmann and Gary Cornell ,Pearson, Eight edition(Unit 5 and Unit 6)

References:

1] The Java Tutorials From ORACLE Java Documentation URL: http://docs.oracle.com/javase/tutorial/ (Refer For All Units)
2]The Java Tutorial: A Short Course on the Basics by Raymond Gallardo, Scott Hommel, Sowmya Kannan, Publisher: Addison-Wesley Professional. (6th Edition)
3]JAVA-The Complete Reference: Herbert Schildt, Oracle Press, Mcgraw Hill,(9th Edition).
4]JAVA™ HOW TO PROGRAM, By Deitel Paul , Deitel Harvey. Publisher: PHI Learning..(10th Edition)
5]Thinking in Java by Bruce Eckel, Prentice Hall,(4th Edition)
6]A Programmer’s guide to JAVA SCJP Certification: Khaleed Mughal and Rolf W. Rasmussen, Addison Wesley, (3rd Edition)

Experiment List

Experiment No. 1: Creating classes and initializing objects.
Aim and Objectives: Applying Classes and Object concepts**.**
Outcomes: Student will be able to use class and objects.
Theoretical Background: classes and objects
Experimentation: Develop a Java Program to implement class and create its objects.
Results and Discussions: Objects are created, data members of class are initialized, members functions can be called.
Conclusion: You can call non-static members of class using objects, constructor get called when you create object of class.

Experiment No. 2: Classes and Objects
Aim and Objectives: Applying Classes and Object concepts**.**
Outcomes: Student will be able to use class, its members and create objects.
Theoretical Background:classes and objects
Experimentation: Develop a Java Program to implement class and create its objects.
Results and Discussions: Objects are created, data members of class are initialized, members functions can be called.
Conclusion: You can call static members of class using class name, static data member is common between all objects.

Experiment No. 3: Implementing multiple Inheritance in Java.

Aim and Objectives: Multiple Inheritance in Java.
Outcomes: Students will be able to create multiple inheritance with the help of Interfaces.
Theoretical Background: Java do not support multiple inheritance with classes, if programmer need to multiple inheritance structure she needs to use interfaces.
Experimentation: Developing Java program with interface inheritance.
Results and Discussions: Interface methods are overridden in class, calling overridden functions using objects is done.
Conclusion: The Class which inherit interface, must have to provide override methods declared in interface. The default access to members of interface is public.

Experiment No. 4:Inheritance in Java
-Aim and Objectives: Inheritance using classes.
Outcomes: Students will be able to create hierarchical inheritance with the help of classes.
Theoretical Background: Child class should extend base class to acquire base class properties.
Experimentation: Create Separate Engine, Tyre, and Door Class. Create a Car class using these classes. And show functionality of each component in the car.
Results and Discussions: When class extends from another class. The extending class is child and the class which is extended is base. The child class object can call public and protected members of base class. Child class can call its own members despite of its access specifier. When you create object of child class its constructor get called. The child class constructor calls the base class constructor.
Conclusion: Child class acquire the properties of base class.

Experiment No. 5: Creating Packages.
Aim and Objectives: Creating JAR(Java Archive)-package.
Outcomes: Student will be able to create deployable packages of Java application.
Theoretical Background: Application need multiple classes. You can not have all classes in single file. So, You have to club together the required classes of your application. This can be done with packages in Java. The archive of these packages is called as JAR. Jar file can be created as executable and non-executable jar files.
Experimentation:-Develop a mathematical package for Statistical operations like Mean, Median, Average, Standard deviation. Create a sub package in the math package -convert. In "convert" package provide classes to convert decimal to octal, binary, hex and vice-versa. Develop application program to use this package, and build executable jar file of it.
Results and Discussions: The default access of class is package. You can create executable version of JAR file.
Conclusion: The applications developed using Java, which contains multiple ".class" files and hierarchical directory structure should be deployed using JAR

Experiment No. 6:Understanding Constructors
Aim and Objectives: Understanding and using parameterized constructors in Java.
Outcomes: Students will be able to use multiple constructor in single class.
Theoretical Background: Java provides zero parameter constructor, program do not need to write it. This is the reason, the zero parameter constructor is called as default constructor. Programmer can write its own constructor which is zero parameter or parameterized constructor. When programmer choose to write

constructor in class, java do not provide default constructor. In this case the programmer has to write zero parameter constructor on its own.

Experimentation: Develop a class Expr to create and evaluate given expression. Constructor accepts the expression as String. For example, Expr("x^2") or Expr("sin(x)+3*x"). If the parameter in the constructor call does not represent a legal expression, then the constructor throws an IllegalArgumentException. The message in the exception describes the error. Provide eval(double num) and eval(int num) method to evaluate given expression and return evaluated answer. For example, if Expr represents the expression 3*x+1, then func.value(5) is 3*5+1, or 16.Finally, getDefinition() returns the definition of the expression. This is just the string that was used in the constructor that created the expression object

Results and Discussions: Calling specific constructor from available ones.
Passing arguments to object. Use of "this" keyword to identify current object class members.

Conclusion: Java class can have overload of constructors, This keyword identifies current object under execution

Experiment No. 7:Exception Handling in Java.

Aim and Objectives: Handling exception in java program.

Outcomes: Students will be able to deal/tackle runtime error in java program.

Theoretical Background: Java have two error reporting mechanisms. One is compile time and one is runtime. The compile time error deals with syntax of Java programming language and you can not get byte code of java program unless your program is error free. The run time error are called as exceptions. They represent semantic issues in program.

Experimentation: Write a class to represent Roman numerals. The class should have two constructors. One constructs a Roman numeral from a string such as "XVII" or "MCMXCV". It should throw a NumberFormatException if the string is not a legal Roman numeral. The other constructor constructs a Roman numeral from an int. It should throw a NumberFormatException if the int is outside the range 1 to 3999.In addition, the class should have two instance methods. The method toString() returns the string that represents the Roman numeral. The method toInt() returns the value of the Roman numeral as an int

Results and Discussions: You can handle exceptions using try, catch blocks, The mission critical code which should execute in the both scenarios- i.e whether exception occurred or application executed normally- should be written in finally block.

Conclusion: Runtime errors can be handled with try, catch and finally block

Experiment No. 8:File Handling

Aim and Objectives: Performing file handling using Java Program.

Outcomes: Student will be able to use I/O streaming classes in Java for file handling.

Theoretical Background: Java provide Input and Output streaming classes. They can be used to deal with input output devices. Here students will learn how to write file operation using Java program.

Experimentation: Take file name as input to your program, If file is existing the open and display contents of the file. After displaying contents of file ask user – do you want to add the data at the end of file. If a user gives yes as response, then accept data from user and append it to file. If file in not existing then create a fresh new file and store user data into it. User should type exit on new line to stop the

program. **Results and Discussions:** Java can deal with file using byte streams and character streams. It have variety of classes to deal with file operation. **Conclusion:** Java application can perform file handling.	
Experiment No. 9: Buffered Streams in Java. **Aim and Objectives:** Using wrapper classes to reduce disk operations in file handling. **Outcomes:** Student will be able to develop efficient file handling programmes. **Theoretical Background:** BufferedReader/Writer, DataInputStream/ OutputStream, PrintWriter are wrapper classes, which improves efficiency of Java program. Student should be able to use buffering of data to avoid disk I/O operations for every read and write of character/word to and from file. **Experimentation:** Take Student information such as name, age, weight, height, city, phone from user and store it in the file using DataOutputStream and FileOutputStream and Retrieve data using DataInputStream and FileInputStream and display the result. **Results and Discussions:** Using buffering for file handling is efficient than byte or character data. **Conclusion:** Students can use buffers to deal with file handling to avoid unnecessary disk access.	
Experiment No. 10: GUI programming in Java **Aim and Objectives:** To learn GUI development packages, creating GUI, handling events. **Outcomes:**Students will be able to develop GUI applications,they should be able to handle events of java components . **Theoretical Background:** **Experimentation:**Develop a Swing GUI based standard calculator program. **Results and Discussions:** **Conclusion**	
Experiment No. 11: Developing Multi-threaded application. **Aim and Objectives:** To learn to create multi threaded applications, Understanding how to create threads. **Outcomes:** Students will be able to execute parts of the program paralley. **Theoretical Background:** **Experimentation:**Write a program that bounces a blue ball inside a JPanel. The ball should begin moving with a mousePressed event. When the ball hits the edge of the JPanel, it should bounce off the edge and continue in the opposite direction. The ball should be updated using a Runnable. **Results and Discussions:** **Conclusion:**	
Experiment No. 12: Networking Application. **Aim and Objectives:** To learn Networking packages in Java. **Outcomes:**Students will be able to use networking packages in Java, using it they can develop network based applications. **Theoretical Background:** **Experimentation:**Write a Swing GUI based network server program. The program is a simple file server that makes a collection of files available for transmission to clients. When the server starts up, it needs to know the name of the directory that contains the collection of files. Specify this directory name through JFileChooser Dialog. You can assume that the directory contains only regular files	

(that is, it does not contain any sub-directories). When a client connects to the server, the server first reads a one-line command from the client. The command can be the string "index". In this case, the server responds by sending a list of names of all the files that are available on the server. Or the command can be of the form "get ", where is a file name. The server checks whether the requested file actually exists. If so, it first sends the word "ok" as a message to the client. Then it sends the contents of the file and closes the connection. Otherwise, it sends the word "error" to the client and closes the connection.

Results and Discussions:

Conclusion:

Experiment No. 13 Database Application.

Theoretical Background: Java language provide sql package. Using this package it is possible to write programs which can interact with database using SQL queries. The driver program is used to convert java input into sql understandable code, the result returned by database can be handled in java program.

Experimentation: Write a GUI based program to create a student registration and Login. Store Registration data in Database and take Login information from Database.

Results and Discussions: Students will be able to use

Conclusion:

Experiment wise Measurable students Learning Outcomes:

Title of the Course: Operating Systems Lab **Course Code: UCSE0533**	**L**	**T**	**P**	**Credit**
	--	--	2	1

Course Pre-Requisite: Fundamentals of Electronics and Computer

Course Description: This course is introduced at third year level to get the students familiar with the basic concepts of computer operating systems in detail.

Course Learning Objectives:

clo1: To make the students understand basic concepts of operating system.

clo2. To expose the students to various functions of the Operating system and their usage.

clo3. To make the students understand CPU scheduling, Memory management and disk scheduling policies

Course Outcomes:

CO	**After the completion of the course the student should be able to**
CO1	Summarize the features of different operating systems
CO2	Experiment with configuration of operating systems
CO3	Compare algorithms for CPU scheduling, Disk Scheduling and page replacement policies.

CO-PO Mapping:

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1	**2**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**1**	**-**
CO2	**2**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**2**	**-**

CO3	**-**	**2**	**-**	**2**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**-**	**2**	**1**

Assessments :

Teacher Assessment:

Two components of In Semester Evaluation (ISE), and one End Semester Examination (ESE) having 20%, 30% and 50% weights respectively.

Assessment	Marks
ISE	50

ISE is based on assignment/declared test/quiz/seminar/Group Discussions etc.

ESE: Assessment is based on 100% course content with60-70% weightage for course content (normally last three modules) covered after MSE.

Course Contents:

Experiment No.1 Installation of Different Operating Systems	**2 Hrs.**
Experiment No.2 Study of MBR and Customizing Grub Loader	**2 Hrs.**
Experiment No.3 Configuring different services for operating system	**2 Hrs.**
Experiment No.4	**2 Hrs.**

Study features of different File Systems	
Experiment No.5 Analyze different CPU scheduling algorithms	**2 Hrs.**
Experiment No.6 Understanding process management using different system calls	**2 Hrs.**
Experiment No.7 Draw Resource Allocation Graph for deadlock detection, and conclude necessary ste avoid it.	**2 Hrs.**
Experiment No.8 Implementation of Banker's Algorithm	**2 Hrs.**
Experiment No.9 Compare different Page Replacement Policies with reference to no. of page faults	**2 Hrs.**
Experiment No.10 Analyze performance of disk scheduling algorithms	**2 Hrs.**
Textbooks: 1. Abraham Silberschatz, Peter Baer Galvin, Greg Gagne: Operating System Principles, 8th edition, Wiley India, 2009	
References: 1. Operating Systems –Concepts and design –Milan Milenkovic (TMGH) (For Types of Operating Systems - Refer Chapter 1 in Operating Systems – Concepts and design – Milan Milenkovic (TMGH)) 2. Operating Systems: Internals and Design Principles (8th Edition)- by William Stallings(Pearson Education International) 3. Modern Operating Systems by Andrew S. Tanenbaum (Pearson Education International)	
Unit wise Measurable students Learning Outcomes:	

Experiment wise Measurable students Learning Outcomes:

1. Students will be able to explain basic concepts of operating systems

2. Students will be able to install Operating System.

3. Students will be able to configure services of Operating system

4. Student will be able to analyze CPU scheduling algorithm

5. Students will be able to detect, avoid and recover deadlock in Operating system

6. Student will be able to analyze page replacement and disk scheduling policies

SEM-II

Title of the Course: Unix Operating Systems	**L**	**T**	**P**	**Credit**
Course Code:UCSE0601	**3**	**-**	**-**	**3**

Course Pre-Requisite: Operating System

Course Description:

This course is introduced at third year level to get the idea of internal working of Unix operating system in detail.

Course Objectives: To expose students to

1. Fundamental architecture of UNIX operating system kernel.

2. Detail algorithms of buffer cache management. 3. Internal File system organizations and related algorithms in UNIX.

4. System calls for UNIX file system.

5. Process structure, creation and management in UNIX.

6. Architecture and algorithms of process scheduling and memory management.

7. I/O subsystem architecture and algorithms.

Course Outcomes: At the end of the course the student will be able to:

CO1	List Features of Unix Operating System.
CO2	Describe Block diagram of Unix Kernel
CO3	Illustrate internal representation of Unix File.
CO4	Compare IPC mechanisms of Unix.

Mapping of course outcomes with program outcomes

COs	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2
CO1	1	-	-	-	-	-	-	-	-	-	-	-	-	-
CO2	1	-	-	-	-	-	-	-	-	-	-	-	1	-
CO3	-	2	-	-	-	-	-	-	-	-	-	-	-	-
CO4	-	2	-	2	-	-	-	-	-	-	-	-	-	1

Assessments:

Teacher Assessment:

Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one End Semester Examination (ESE) having 20%, 30% and 50% weights respectively.

Assessment	Marks
ISE 1	10
MSE	30
ISE 2	10
ESE	50

ISE 1 and ISE 2 are based on assignment/declared test/quiz/seminar/Group Discussions etc.

MSE: Assessment is based on 50% of course content (Normally first three modules)

ESE: Assessment is based on 100% course content with 60-70% weightage for course content (normally last three modules) covered after MSE.

Course Contents:

Unit 1:--- Introduction and buffer cache: General Overview of the System - History, System Structure, User Perspective, Operating System Services, Assumption About Hardware, Architecture of UNIX OS, Introduction to system concepts, Kernel Data Structure, System Administration. Buffer Cache: - Buffer headers, structure of the buffer pool, scenarios for retrieval of a buffer, reading and writing disk blocks, advantages and disadvantages of cache.	**8 Hrs.**

Unit 2:---Internal Representation of Files I-nodes, structure of the regular file, directories, conversion of a pathname to i-node, super block, i-node assignment to a new file, allocation of disk blocks, other file types.	**5 Hrs.**
Unit 3:--System Calls for file system: System Calls for file system:- Open, Read, write, File and Record Locking, Adjusting the position of FILE I/O-LSEEK, Close, File Creation, Creation of Special File, Change Directory and Change Root, Change Owner and Change Mode, Stat and fstat, Pipes, Dup, Mounting and Un-mounting file systems, Link, Unlink, File System Abstractions, File system maintenance.	**6 Hrs.**
Unit 4:--- The Structure of process: Process stages and transitions, layout of system memory, the context of a process, Saving context of a process, manipulation of the process address space.	**6 Hrs.**
Unit 5:---Process Control and Scheduling: Process Control: - Process creation, signals, process termination, awaiting process termination, invoking other programs, the user id of a process, the shell, System Boot and the Init process. Process Scheduling: - Process Scheduling, system call for time, clock.	**6 Hrs.**
Unit 6:--- Memory management and I/O Subsystem: Swapping, Demand passing, a hybrid system with demand paging and swapping. Driver interfaces, disk drives, terminal drivers, Streams.	**6 Hrs.**
Textbooks:1. The Design of Unix Operating System - Maurice J. Bach (PHI)	
References:1. Linux System Programming - Robert Love, Publisher - SPD, O’ REILLY 2. Unix concepts and administration – 3rd Edition – Sumitabha Das (TMGH).	

	L	T	P	Credit
Title of the Course: Compiler Construction	**L**	**T**	**P**	**Credit**
Course Code: UCSE0602	**3**	**1**		**4**

Course Pre-Requisite: Automata Theory, Data Structures

Course Description: This course explores the principles, algorithms, and data structures involved in the design and construction of compilers. Topics include Language processors, lexical analysis, context-free grammars, push-down parsers, LR and LALR parsers, other parsing techniques, symbol tables and introduction to intermediate code generation.

Course Objectives:

1. To expose the students to the fundamentals of languages processing and Language Processors.
2. To make students to learn design of grammars, assemblers, parsers and compilers.
3. To design and implement various phases of compiler.

Course Learning Outcomes:

CO	**After the completion of the course the student should be able to**
CO1	Explain basics of Languages and Language processors for the System programming.
CO2	Analyze phases and steps for Software Program Execution in detail from Analysis to Execution.
CO3	Explain different phases of compiler in detail.
CO4	Design and develop modules for different phases of Compiler.

CO-PO Mapping:

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
CO1	3													
CO2		2												
CO3		3											3	
CO4			3		3								2	

Assessments :

Teacher Assessment:

Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one End Semester Examination (ESE) having 20%, 30% and 50% weights respectively.

Assessment	Marks
ISE 1	10
MSE	30
ISE 2	10
ESE	50

ISE 1 and ISE 2 are based on assignment/declared test/quiz/seminar/Group Discussions etc.

MSE: Assessment is based on 50% of course content (Normally first three modules)

ESE: Assessment is based on 100% course content with60-70% weightage for course content (normally last three modules) covered after MSE.

Course Contents:

Unit 1: Assembler & Macros : Introduction, language processing activities, Fundamentals of language processing, Fundamentals of language Specification Assemblers: Elements of assembly language programming, a simple assembly scheme, pass structure of assemblers, design of a two pass assembler, Macro definition and call, Macro expansion, Nested macro calls, Design of macro pre-processor.	**07 Hrs.**
Unit 2: Phases in Compilers-Lexical Analysis& Syntax Analysis: Role of a Lexical analyzer, input buffering, specification and recognition of tokens, Role of Parser, Writing grammars for context free environments, Top-down parsing- Recursive descent and predictive parsers (LL), Bottom-Up parsing- Operator precedence parsing, LR, SLR and LALR parsers.	**08 Hrs.**
Unit 3: Syntax Directed Translation and Intermediate Code Generation: Syntax directed definitions, construction of syntax tree, S-attributed definitions, L-attributed definitions, Intermediate languages, assignment statements, back patching, procedure calls.	**07 Hrs.**
Unit 4: Code Optimization: Sources of optimization, Peephole optimization and basic blocks, loops in flow graphs, Data flow analysis and equations, code improving transformation and aliases	**07 Hrs.**
Unit 5: Code Generation: Issues in design of a code generator and target machine, Run time storage management, Basic blocks and flow graphs, Next use information and simple code generator, Issues of register allocation, code generation from Dags.	**07 Hrs.**
Unit 6: Linker & Loader: Relocation and linking concepts, design of a linker, Self-relocating programs, Loaders**.**	**06 Hrs.**
Textbooks: 1. D.M. Dhamdhere, “Systems Programming and Operating Systems” Second revised Edition, 2005, Tata McGraw- Hill Publishing Company limited, New Delhi. 2. “Compilers - Principles, Techniques and Tools”, A.V. Aho, R. Shethi and J.D. Ullman, Pearson Education.	
References: 1. System Programming -- J. J. Donovan (Mc-Graw Hill). 2. “Compilers - Principles, Techniques and Tools”, A.V. Aho, R. Shethiand J.D. Ullman, Addison Wesley Publishing Company.	

Title of the Course: Information System Security	**L**	**T**	**P**	**Credit**
Course Code: UCSE0603	3	--	--	3

Course Pre-Requisite: Computer Network, Data Communication, Discrete Mathematics

Course Description: This course gives you practical survey of both the principles and practice of cryptography and network security. In the first part of course, the basic issues to be addressed by a network security capability are explored by providing a tutorial and survey of cryptography and network security technology. The latter part of course deals with the practice of network security: practical applications that have been implemented and are in use to provide network security.

Course Objectives:

1. Explain different types of symmetric and asymmetric security techniques
2. Compare different types of cryptographic algorithms to ensure data integrity
3. Explain different types of security protocols in TCP/IP protocol suite
4. Compare different types of techniques used for distribution of secret keys
5. Explain different types of security threats for computer system

Course Learning Outcomes:

CO	**After the completion of the course the student should be able to**
CO1	Explain the use of Cryptographic algorithms to ensure data protection and integrity
CO2	Illustrate the different Network and Internet security protocols in TCP/IP stack
CO3	Apply the knowledge of cryptographic techniques to solve the problems on security
CO4	Analyze the security facilities designed to provide System security

CO-PO Mapping:

CO	**PO 1**	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**	**PS O1**	**PS O2**
CO1	**2**													
CO2	**1**													
CO3	**2**	**3**	**2**	**1**										1
CO4			**2**			**1**		1					2	2

Assessments :

Teacher Assessment:

Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one EndSemester Examination (ESE) having 20%, 30% and 50% weights respectively.

Assessment	Marks
ISE 1	10
MSE	30
ISE 2	10
ESE	50

ISE 1 and ISE 2 are based on assignment/declared test/quiz/seminar/Group Discussions etc.

MSE: Assessment is based on 50% of course content (Normally first three modules)

ESE: Assessment is based on 100% course content with60-70% weightage for course content (normally last three modules) covered after MSE.

Course Contents:

Unit 1: Introduction to Information Security **Overview: (2)** Computer Security Concepts, The OSI Security Architecture, Security Attacks, Security Services, Security Mechanisms, A Model for Network Security **Classical Encryption Techniques: (3)** Symmetric Cipher Model, Substitution Techniques, Transposition Techniques, Rotor machines, Steganography	**05Hrs.**
Unit 2: Symmetric and Asymmetric Key Cryptography **Block Ciphers and the Data Encryption Standard (4)** Block Cipher Structure, Data Encryption Standard (DES), A DES Example, Strength of DES, Block Cipher Design Principles, AES Structure, Multiple Encryption and Triple-DES **Public Key Cryptography (4)** Principles of Public-Key Cryptosystems, RSA Algorithm, Other Public key Cryptosystems - Diffie-Hellman Key Exchange, ElGamal Cryptographic system	**08 Hrs.**
Unit 3: Cryptographic Authentication Functions **Cryptographic Hash Functions: (3)** Applications of Cryptographic Hash Functions, Two Simple Hash Functions, Requirements and Security, Hash Functions Based on Cipher Block Chaining, Secure Hash Algorithm (SHA) **Message Authentication Codes: (3)** Message Authentication Requirements, Message Authentication Functions, Requirements for MAC and Security of MACs, MACs Based	**08 Hrs.**

on Hash Functions: MAC, MACs Based on Block Ciphers: DAA and CMAC **Digital Signatures: (2)** Digital Signatures, ElGamal Digital Signature Scheme, Schnorr Digital Signature Scheme, Digital Signature Standard (DSS)	
Unit 4: Key Management and User Authentication **Key management (3)** Symmetric Key Distribution Using Symmetric Encryption, Symmetric Key Distribution Using Asymmetric Encryption, Distribution of Public Keys, X.509 Certificates, Public Key Infrastructure **User Authentication Protocol (3)** Remote User-Authentication Principles, Remote User-Authentication Using Symmetric Encryption, Kerberos, Remote User Authentication Using Asymmetric Encryption.	**06 Hrs.**
Unit 5: Internet Security Protocols **Transport-Level Security (2)** Web Security Issues, Secure Sockets Layer (SSL), Transport Layer Security (TLS), HTTPS, SSH **Electronic Mail Security (3)** Pretty Good Privacy (PGP), S/MIME, SET **IP Security (2)** IP Security Overview, IP Security Policy, Encapsulating Security Payload	**07Hrs.**
Unit 6: System Security **Intruders (2)** Intrusion Detection, Password Management **Malicious Software (2)** Viruses and Related Threat, Countermeasures, DoS **Firewalls (2)** Firewall Design Principles, Trusted Systems	**06 Hrs.**
Textbooks: 1. Williams Stallings – Cryptography and Network Security Principles and Practices Pearson Education (LPE), 6th Edition and 4th Edition(For Unit 6)	
References: 2. Cryptography & Network Security B.A. Forouzan McGrawHill 3. Cryptography and network security – Atul Kahate (TMGH) 4. Handbook of Applied Cryptography - Menezes, an Oorschot, and S.A. Vanstone	
Unit wise Measurable students Learning Outcomes: **Unit 1 : Overview and Classical Encryption** UO-1.1) Define security attacks, services and mechanisms UO-1.2) Enlist different components of model for network security UO-1.3) Solve numerical based on types of encryption technique for different input UO-1.4) Explain Rotor machines and Steganography	

Unit 2: Block Ciphers and Public Key Cryptography

UO-2.1) Describe encryption and decryption operation of DES and AES

UO-2.2) Differentiate secret key and public-key cryptography

UO-2.3) Solve numerical based on RSA algorithm and Diffie-Hellman key exchange

UO-2.4) Solve numerical based on ElGamal algorithm

Unit 3: Cryptographic Authentication Functions

UO-3.1) Describe different types of cryptographic hash function

UO-3.2) Describe different types of MAC function

UO-3.3) Describe Digital signature standard

Unit 4: Key Management and Distribution

UO-4.1) Explain different schemes of key distribution

UO-4.2) Recognize X.509 security certificate of any website

UO-4.3) Explain public key infrastructure

UO-4.3) Explain User Authentication Protocols

Unit 5: Internet Security Protocols and Applications

UO-5.1) Describe Wireless Network Security

UO-5.2) Describe Transport Layer Security(TLS)

UO-5.3) Describe Web Security

UO-5.4) Describe IP Security

Unit 6: System Security

UO-6.1) Describe Intruders and IDS

UO-6.2) Describe different types of malicious software

UO-6.3) Describe firewall and its different types

Title of the Course: Professional Elective-II-Software Testing and Quality Assurance **Course Code: UCSE0621**	**L**	**T**	**P**	**Credit**
	3	--	--	3

Course Pre-Requisite: Software Engineering

Course Description: Technical and managerial aspects of testing and quality assurance in software development with emphasis on practical projects. Tools for testing and QA. Performance evaluation, verification and validation of software. Managing and documenting software testing and QA activities in local and outsourced projects.

Course Learning Objectives:

1. To provide knowledge about fundamentals of software testing and software quality
2. To understand the fundamentals of software verification
3. To understand and evaluate metrics and models used in software testing
4. To understand and compare testing web applications and desktop applications
5. To understand, compare and Choose from various software project assessment methods

Course Outcomes:

CO	**After the completion of the course the student should be able to**
CO1	Explain basic concepts of Software Testing and Software verification models.
CO2	Identify Test cases and explain SRS and various technical documents.
CO3	Interpret appropriate metrics for evaluation of software projects.
CO4	Explain the concepts of web application testing

CO-PO Mapping:

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
CO1	2													
CO2	2													
CO3	2	2												
CO4	2	2										2	2	

Assessments :

Teacher Assessment:

Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one End Semester Examination (ESE) having 20%, 30% and 50% weights respectively.

Assessment	Marks
ISE 1	10
MSE	30
ISE 2	10
ESE	50

ISE 1 and ISE 2 are based on assignment/declared test/quiz/seminar/Group Discussions etc.

MSE: Assessment is based on 50% of course content (Normally first three modules)

ESE: Assessment is based on 100% course content with60-70% weightage for course content (normally last three modules) covered after MSE.

Course Contents:	
Unit 1. Introduction :Some Software Failures, Testing Process, Some Terminologies, Limitations of Testing, The V Shaped software life cycle model	**4Hrs.**
Unit 2. Software Verification: Verification Methods, SRS document verification, SDD document verification, Source code reviews, User documentation verification, Software project audit Creating test cases from SRS and Use cases: Use Case Diagram and Use Cases, Generation of test cases from use cases, Guidelines for generating validity checks, strategies for data validity, Database testing	**8Hrs.**
Unit 3. Regression Testing: What is regression testing?, Regression Test cases selection, Reducing the number of test cases, Risk analysis, Code coverage prioritization techniques Object oriented testing: What is Object orientation?, What is object oriented testing?, Path testing, State based testing, Class testing	**7Hrs.**
Unit 4. Measurement - what is it and why do it?: Measurement in everyday life, Measurement in software engineering, scope of software Metrics and Models in Software testing: Software Metrics, Categories of Metrics, Object oriented Metrics used in testing, what should we measure during testing, Software Quality attributes prediction models	**6Hrs.**
Unit 5. Measuring Internal Product: Attribute Size, Aspects of software size, Length, Reuse, Functionality Measuring External product Attributes: Modeling software quality, measuring aspects of software quality.	**5Hrs.**
Unit 6. Testing Web applications: What is web testing? Functional testing, UI testing, Usability testing, configurations and compatibility testing, security testing, performance testing, database testing, post deployment testing, web metrics. Automated Test data generation: Automated Test Data generation Approaches to test data generation, Test data generation tools.	**6Hrs.**
Textbooks: 1. Software testing: Yogesh Singh, Cambridge University Press, First Edition 2. Software Metrics – A Rigorous & Practical approach: Norman Fenton, Shari Lawrence Pfleeger, 2nd Edition Thomson Press 3. Software Quality Engineering, Jeff Tian, Wiley India Ltd.	
References: 1. Foundations of Software testing: Aditya P. Mathur, Pearson, Second Edition 2. Software Testing: Ron Patton, Pearson (SAMS), Second Edition 3. Software Quality, Mordechai Ben Menachem, Garry S. Marliss, BS Publications	

Title of the Course: Data Analytics	L	T	P	Credit
Course Code: UCSE0622	**3**	**0**	**0**	**3**

Course Pre-Requisite: data warehousing, basics of mining techniques

Course Description:

Course Objectives:

1. Understand Business Intelligence, decision support systems in Data warehouse.
2. Study the statistical Data analysis techniques for data preparation and exploration.
3. Use data mining tasks for performing data analysis

Course Learning Outcomes:

CO	After the completion of the course the student should be able to
CO1	Demonstrate the knowledge of statistical data analysis used in data analytics.
CO2	Identify appropriate machine learning based data analytics techniques.
CO3	Analyze supervised and unsupervised learning techniques for data analysis

CO-PO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO 1	PSO 2
CO1	-	2	-	-	-	-	-	-	-	-	-	-	-	-
CO2	1	-	1	-	2	-	-	-	-	-	-	1		-
CO3	-	2	-	-	3	-	-	-	-	-	-	-	2	-

Assessments :

Teacher Assessment:

Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one End Semester Examination (ESE) having 20%, 30% and 50% weights respectively.

Assessment	Marks
ISE 1	10
MSE	30
ISE 2	10
ESE	50

ISE 1 and ISE 2 are based on assignment/declared test/quiz/seminar/Group Discussions etc.

MSE: Assessment is based on 50% of course content (Normally first three modules)

ESE: Assessment is based on 100% course content with60-70% weightage for course content (normally last three modules) covered after MSE.

Course Contents:

UNIT-I : Introduction: Business intelligence, Decision Support System and	**04 Hrs.**

Data mining	
UNIT-II: Pre-processing of Data : Data validation, Data transformation , Data reduction, Normalization Techniques	**08 Hrs**
UNIT-III: Inferential Statistics : Inferential Statistics through hypothesis tests **Regression & ANOVA :** Regression, ANOVA(Analysis of Variance)	**06Hrs.**
UNIT-IV: Machine Learning: Regression: Lasso Regression, K Nearest Neighbors, Linear Regression, Logistic Regression	**06 Hrs.**
UNIT-V: Supervised Learning techniques : Regression and Classification Trees, Support Vector Machines, Ensemble Methods: Random Forest, Decision trees, K Means Clustering Algorithms, Artificial Neural Networks, Deep learning	**08 Hrs.**
UNIT-VI: Unsupervised Learning techniques: Clustering, Associative Rule Mining, Challenges in big data analytics **Reinforcement Learning**	**06 Hrs.**
Textbooks: 1. Business Intelligence- Data Mining and optimization for Decision Making- Carlo Vercellis- Wiley Publications. 2. Data mining Introductory and Advanced topics- Margaret H. Dunham- Pearson 3 James, G., D. Witten, T. Hastie, and R. Tibshirani, An Introduction to Statistical learning with Application to R, Springer, New York. 2013	
References: 1. Data Mining: Concepts and Techniques Second Edition- Jiawei Han and Micheline Kamber- Morgan KaufMan Publisher. 2. DATA MINING AND ANALYSIS Fundamental Concepts and Algorithms- MOHAMMED J. ZAKI and WAGNER MEIRA JR.- Cambridge University Press .	

Title of the Course: Adhoc Wireless Networks	**L**	**T**	**P**	**Credit**
Course Code: UCSE0623	**3**	**0**	**0**	**3**

Course Pre-Requisite: Basic computer network concepts.

Course Description: Adohc wiresless netwroks subject deals with styudy of architecture of wireless network infrastructure, protocols and its services.

Course Objectives:

1. To explain fundamental principles of Ad-hoc Networks
2. To discuss a comprehensive understanding of Ad-hoc network protocols
3. To outline current and emerging trends in Ad-hoc Wireless Networks.
4. To analyze energy management in ad-hoc wireless networks.

Course Learning Outcomes:

CO	**After the completion of the course the student should be able to**
CO1	compare the differences between cellular and ad hoc networks and analyze the challenges at various layers and applications.
CO2	summarize the protocols used at the MAC layer and scheduling mechanisms.
CO3	compare and analyze types of routing protocols used for unicast and multicast routing.
CO4	examine the network security parameters and routing mechanism.
CO5	evaluate the energy management schemes and Quality of service parameters in ad hoc networks.

CO-PO Mapping:

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
CO1	2	2											1	
CO2	1	1											1	
CO3	2	2											1	
CO4	1	2											1	
CO5	1	2												

Assessments :

Teacher Assessment:

One component of In Semester Evaluation (ISE) and one End Semester Examination (ESE) having 50%, and 50% weights respectively.

Assessment	Marks
ISE	50
ESE	50

ISE are based on practical performed/ Quiz/ Mini-Project assigned/ Presentation/ Group Discussion/ Internal oral etc.

ESE: Assessment is based on oral examination

Course Contents:

UNIT-I : Ad-hoc wireless networks and MAC Protocol Ad-hoc Wireless Networks Introduction, Issues in Ad-hoc Wireless Networks, Ad-hoc Wireless Internet; Cellular and Ad Hoc wireless networks, Applications, Issues in Ad Hoc wireless networks.	**06Hrs.**

UNIT-II: MAC Protocols for Ad-hoc wireless networks: MAC Protocols for Ad-hoc Wireless Networks: Introduction, Issues in Designing a MAC Protocol, Design Goals of MAC Protocols, Classification of MAC protocols, Contention-Based Protocols, Contention-Based Protocols with Reservation Mechanisms, Contention-Based Protocols with Scheduling Mechanism	**08 Hrs**
UNIT-III: Routing protocols in Adhoc wireless protocol Introduction, Issues in designing a routing protocol for ad hoc wireless networks, Classification of routing protocols, Table driven protocols :-DSDV, WRP, CGSR; On-Demand Hybrid routing protocols:-DSR, AODV, LAR, ABR, SSA , ZRP, ZHLS	**08 Hrs.**
UNIT-IV: Multicast Routing for Ad hoc wireless networks Introduction, Issues in designing a multicast routing protocol, Operation of multicast routing protocols, An architecture reference model for multicast routing protocols, Classification of multicast routing protocols, Tree-based Multicast Routing Protocols:-BEMR, MZRP, MAODV ; Mesh-based multicast routing protocols:-NSMP, CAMP	**06 Hrs.**
UNIT-V: Transport Layer and Security Protocols Transport Layer and Security Protocols for Ad-hoc Networks: Introduction, Issues in Designing a Transport Layer Protocol; Design Goals of a Transport Layer Protocol TCP over adhoc wireless networks:-TCP-F,Ad Hoc TCP, Split TCP; **Security in ad hoc wireless networks:** Network security requirements, Issues and challenges in security provisioning, Network security attacks,Secure routing protocol - SAR, Security**-** Aware AODV Protocol.	**08 Hrs.**
UNIT-VI:Energy Management and Quality of service Introduction, Need, Classification of energy management schemes, System Power Management schemes-Processor Power Management Scheme, Device Power Management Scheme. Issues and challenges, Classification of QoS colutions, QoS framework – INSIGNIA, INORA, SWAN.	**06 Hrs.**
Textbooks: 1. C. Siva Ram Murthy & B. S. Manoj: Ad-hoc Wireless Networks, 2nd Edition, Pearson Education, 2011	
References: 1] C.K. Toh: Ad-hoc Mobile Wireless Networks -Protocols and Systems, Pearson Education, 2002 2] Ozan K. Tonguz and Gianguigi Ferrari: Ad-hoc Wireless Networks, John Wiley, 2007.	

Title of the Course: Cyber Security Course Code: UOEL0611	L	T	P	Credit
	3	-	-	3

Course Pre-Requisite:

1. Basics of Computer System 2. Hands on Internet Applications

Course Description: Cyber Security is designed to help learners develop a deeper understanding of modern information and system protection technology and methods. We hope learners will develop a lifelong passion and appreciation for cyber security, which we are certain, will help in future endeavours. Students, developers, managers, engineers, and even private citizens will benefit from this learning experience. All aspects of security are covered in this course.

Course Objectives:

1. To make students familiar with basics of Information system components
2. To make students familiar with Information Security and demonstrate security threats
3. To make students be able to develop strategies for developing secure Information System.
4. To make aware students about security issues in Computer and Network Hardware
5. To make students aware of security policies and Information Security

Course Learning Outcomes:

CO	After the completion of the course the student should be able to
CO1	Define the basic components of Information system and understand development issues.
CO2	Explain Information Security and different types of threats to Information System.
CO3	Assess aspects of Information System with respect to security policies
CO4	Explain security issues in computer and network Hardware
CO5	Develop various security policies and select Information Security Standard for real world Information system

CO-PO Mapping:

CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO1	PSO2
CO1	3	2	-	-	-	-	-	-	-	-	-	-	-	-
CO2	3	-	-	3	-	-	-	-	-	-	-	-	1	-
CO3	-	-	-	3	-	-	-	-	-	-	-	-	-	-
CO4	-	-	2	-	-	-	-	-	-	-	-	-	-	-
CO5	-	2	-	-	-	-	-	-	-	-	-	-	-	2

Assessments :

Teacher Assessment:

Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one EndSemester Examination (ESE) having 20%, 30% and 50% weights respectively.

Assessment	Marks
ISE 1	10
MSE	30
ISE 2	10
ESE	50

ISE 1 and ISE 2 are based on assignment/declared test/quiz/seminar/Group Discussions etc.

MSE: Assessment is based on 50% of course content (Normally first three modules)

ESE: Assessment is based on 100% course content with60-70% weightage for course content (normally last three modules) covered after MSE.

Course Contents:

Unit 1: Introduction to Information Systems Introduction, Modelling the Business System, Information System Components and Categories, Individuals in Information System, Development of Information System.	05 **Hrs.**
Unit 2: Information Security **Information Security** Introduction, Threats to Information System, Information Assurance, Cyber Security and Security Risk Analysis. **Application Security:** Introduction, Data Security consideration, Security Technology, Intrusion Detection and Access Control.	08 **Hrs.**
Unit 3: Security Threats Introduction to Security Threats, Network and Services Attacks, Security Threats to E-commerce.	08 **Hrs.**
Unit 4: Development of Secured Information System Introduction, Developing Secured Information System, Key elements of Information Security Policies, Information System Development Life Cycle, Application Security, Information Security Governance and Risk Management, Risk Management, Security Architecture and Design.	06 **Hrs.**
Unit 5: Security Issues in Hardware: Introduction, Data Storage and Downloadable devices, Physical Security of IT Assets, CCTV and Intrusion Detection System, Security Measures.	05 **Hrs.**
Unit 6: Security Policies and Information Security Standards: **Security Policies:** Introduction, Why do we need Security Policies ?, Security Policy Development, E-mail Security Policies, Policy Review Process, Corporate Policy, Sample Template of Cyber Security Policy. **Information Security Standards:** Introduction, IT Act 2000, Copyright, Patent, Intellectual Property Rights, Cyber Laws in India, Software Licensing, Semiconductor Laws and Patent Law.	08 **Hrs.**

Textbooks:

1. Fundamentals of Cyber Security, By Mayank Bhushan, Rajkumar Rathore and Aatif Jamshed, BPB Publications.

References:

1. Data communication and Networking by Behrouz A. Forouzan, McGraw Hill Education (India) Pvt. Ltd.
2. Certified Ethical Hacker Certification Exam by William Manning.

Title of the Course: Joy of Computing **Course Code:** UOEL0612	**L**	**T**	**P**	**Credit**
	3	-	-	3

Course Pre-Requisite:

1. Basics knowledge of computers

Course Description: This course intents to inspire students to use programming to solve computational problems. As a part of this course students will learn and practice programming skills using python.

Course Learning Objectives:

1. Provide an understanding of the role computation can play in solving problems.
2. Help students who do not have background of Computer Science and Engineering to feel confident of their ability to write small programs that allow them to accomplish useful goals.

Course Outcomes:

CO	**After the completion of the course the student should be able to**
CO1	Recall the basics concept of Python programming.
CO2	Summarize different decision control and looping constructs in python
CO3	Demonstrate concepts of Object Oriented Programming
CO4	Write python programs related to simple-moderate mathematical/logical problems.

CO-PO Mapping:

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PSO 2**
CO1														
CO2														
CO3						1	1							
CO4					1	1	1					1	1	1

Assessments :

Teacher Assessment:

Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one End Semester Examination (ESE) having 20%, 30% and 50% weights respectively.

Assessment	Marks
ISE 1	10
MSE	30
ISE 2	10
ESE	50

ISE 1 and ISE 2 are based on assignment/declared test/quiz/seminar/Group Discussions etc.
MSE: Assessment is based on 50% of course content (Normally first three modules)

ESE: Assessment is based on 100% course content with60-70% weightage for course content (normally last three modules) covered after MSE.

Course Contents:	
Unit 1: Introduction: What is Computing? Setting up the programming environment of python- Anaconda, Spider IDE, writing basic program and executing it	07 **Hrs.**
Unit 2: Basics of Python: Variables and Expressions, Loops and Conditional statements in python Decomposition, Abstractions, Functions	04 **Hrs.**
Unit 3: Data Structure in python: Lists, Tuples and Dictionaries, Developing sorting techniques of data structure in python, Searching for elements in List, Tuples, Recursion: Writing recursive function calls in python	08 **Hrs.**
Unit 4: Object Oriented Programming: Why to use OOP for modelling the problem? What is class? What is Object?	04 **Hrs.**
Unit 5: Python Classes and Inheritance abstract data types through classes, getters and setters, information hiding, class variables, Inheritance	08 **Hrs.**
Unit 6: Introduction to File Handling File Built-in function, File Built-in methods, File Built-in Attributes, standard files and command line arguments.	05 **Hrs.**

Textbooks:1. Wesley J Chun,"Core Python Programming", Second Edition, Pearson Publication

References:

1. Introduction to Computer Science and Programming in Python:- MIT Open Courseware
2. JOC using Python-NPTEL course

Unit wise Measurable students Learning Outcomes:

Title of the Course: Information System Security Lab **Course Code:** UCSE0631	**L**	**T**	**P**	**Credit**
	0	**0**	**2**	**1**

Course Pre-Requisite: Computer Network, Programming languages like C,C++ and Java

Course Description: This course is to designed to do the practical implementation of Cryptographic algorithms and have the hands-on experience on open source tools available to demonstrate the security concepts.

Course Objectives:

1. To study the security services, mechanism and attacks
2. To acquire the knowledge of different cryptographic algorithms and its uses
3. To gain the practical experience of using the security tools for various applications

Course Learning Outcomes:

CO	**After the completion of the course the student should be able to**
CO1	Demonstrate encryption and authentication mechanisms
CO2	Implement various cryptographic algorithms
CO3	Make use of various security tools to analyze the security concepts

CO-PO Mapping:

CO	**PO 1**	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**	**PSO 1**	**PS O2**
CO1	2			1										
CO2		2		2									2	
CO3			2		3			2					2	

Assessments :

Teacher Assessment:

One component of In Semester Evaluation (ISEI) and one In Semester Evaluation (ISEII) having 33%, and 67% weights respectively.

Assessment	Marks
ISEI	25
ISEII	50

ISEI are based on practical performed/ Quiz/ Mini-Project assigned/ Presentation/ Group Discussion/ etc.

ISEII: Internal Oral

Course Contents:

Experiment No. 1: Classical Encryption techniques-Caesar Cipher **02 Hrs.**

Aim and Objectives: To implement the program on Substitution ciphers like Caesar Cipher, Playfair Cipher, Hill Cipher etc

Theoretical Background: Substitution Ciphers-Caesar cipher

Experimentation: 1) Perform Encryption using Substitution ciphers like Caesar Cipher, Playfair Cipher, Hill Cipher , Vigenere cipher etc

2) Perform Decryption function for given input

Experiment No. 2: Classical Encryption techniques-Vigenere Cipher

Aim and Objectives: To implement the program on Substitution ciphers like Caesar

Cipher, Playfair Cipher, Hill Cipher etc **Theoretical Background:** Substitution Ciphers-Caesar cipher **Experimentation**: 1) Perform Encryption using Substitution ciphers like Caesar Cipher, Playfair Cipher, Hill Cipher , Vigenere cipher etc 2) Perform Decryption function for given input	
Experiment No. 3: Classical Encryption techniques-Transposition Ciphers **Aim and Objectives:** To implement the program on Transposition ciphers like Rail fence technique, Columnar transposition etc **Theoretical Background:** Rail fence technique **Experimentation:** Given a plain-text message and a numeric key, cipher/de-cipher the given text using Rail Fence algorithm.	**02Hrs.**
Experiment No.4: Symmetric Ciphers-DES **Aim and Objectives:** Implement a program to perform Encryption and Decryption on DES cipher in Java **Theoretical Background:** DES Cipher **Experimentation:** 1) Take input as Plaintext block – text or stream of bits 2) Generate the key for Encryption of plaintext 3) Perform Encryption using DES Cipher functions 4) Perform DES Decryption	**02Hrs.**
Experiment No. 5: Asymmetric Ciphers- RSA algorithm **Aim and Objectives:** To implement a program to show encryption and decryption using RSA algorithm in Java/C/C++ **Theoretical Background:** RSA algorithm **Experimentation:** 1) Input : Plaintext (text or integer) 2) Write the Encryption function of RSA to generate Ciphertext 3) Write the decryption function of RSA to obtain plaintext from the Ciphertext	**02Hrs.**
Experiment No. 6: **Public key Cryptosystem-Diffie Hellman Algorithm** **Aim and Objectives:** To implement a program on Diffie Hellman key exchange algorithm in Java **Theoretical Background:** Diffie Hellman Key exchange algorithm **Experimentation:** 1) Calculate Public keys from private keys for the users 2) Calculate session key from the public key using DH algorithm	**02Hrs.**
Experiment No. 7: Hash functions **Aim and Objectives:** To implement the program on Hash functions –SHA, MD5 etc to show the integrity check on the files transferred **Theoretical Background:** Hash functions – SHA, MD5 **Experimentation:** 1.To demonstrate the use of Hash functions in Linux and Windows(Using commands) 2. To use various online tools to generate the Hash values 3. Implement the program to generate Hash value using Hash functions in Java	**02Hrs.**
Experiment No. 8: Creation of Digital Signature **Aim and Objectives:** To create the digital signature for documents **Theoretical Background:** Digital Signature. **Experimentation: 1)** Create Digital Signature for Adobe reader documents /other applications by implementing security settings in the software	**02Hrs.**
Experiment No. 9: Creation of Digital certificate **Aim and Objectives:** To create a self signed certificate using java KeyTool **Theoretical Background:** Digital Certificate – X.509, SSL certificate **Experimentation:** 1) Generate self-signed certificate using Java Keytool command 2) Install Self signed certificate in Windows	**02Hrs.**

3) Demonstrate the use of certificate	
Experiment No. 10:Demonstration of SSL protocol **Aim and Objectives:** To demonstrate the working of SSL protocol using Network analyzer tools like Wireshark **Theoretical Background:** SSL/TLS security protocol. **Experimentation:1)** Install Wireshark or any other Network Analyzer tool on Windows **2)** Perform Network analysis	**02Hrs.**
Experiment No. 11: Use of Security tools **Aim and Objectives:** To demonstrate the use of open security tools in Kali linux etc to perform ethical hacking **Theoretical Background:** Ethical Hacking **Experimentation:** Demonstrate the use of open source security tools like Wireshark, Nesuss, N.Map, Zen-Map etc for different attacks	**02 Hrs.**
Experiment No. 12: **Aim and Objectives: Case study of Cyber attack/Seminar on the Cyber attack** **Theoretical Background:** Cyber security **Experimentation:1**) To obtain the information on various cyber attacks 2) Provide the case study and prepare the questionnaire explaining the cyber attack	**02 Hrs.**
Textbooks: 1. Williams Stallings – Cryptography and Network Security Principles and Practices Pearson Education (LPE), 6th Edition	
References: 1. Cryptography & Network Security B.A. Forouzan McGrawHill 2. Cryptography and Network security – Atul Kahate (TMGH) 3. Handbook of Applied Cryptography - Menezes, an Oorschot, and S.A. Vanstone	
Experiment wise Measurable students Learning Outcomes: 1. To acquire knowledge of Cryptography and Network Security 2. To apply the security techniques, Protocols and Algorithms 3. To demonstrate the use of security tools to analyze the security concepts	

Title of the Course: Software Testing & Quality Assurance Lab	**L**	**T**	**P**	**Credit**
Course Code: UCSE0633	**0**	**0**	**2**	**1**

Course Pre-Requisite: Software Engineering, OOMD

Course Description: This course is to do the practical implementation of verification, Validation/Testing and Quality Assurance concepts.

Course Objectives:

1. To study the verification techniques.
2. To acquire the knowledge of different testing types.
3. To gain the practical experience of using the verification/validation concepts for quality assurance.

Course Learning Outcomes:

CO	**After the completion of the course the student should be able to**
CO1	Build Verification document for SRS, SDD and Code Document.
CO2	Plan and Construct Test plan and test cases for SRS, Use Cases
CO3	Make use of Automation Tools for testing various types of S/W applications.

CO-PO Mapping:

CO	**PO1**	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**	**PSO2**
CO1	2								2					
CO2	2	2	2						2			1	2	
CO3		2			3			1				2	2	

Assessments :

Teacher Assessment:

One component of In Semester Evaluation (ISEI) and one In Semester Evaluation (ISEII) having 33%, and 67% weights respectively.

Assessment	Marks
ISEI	25
ISEII	50

ISEI are based on practical performed/ Quiz/ Mini-Project assigned/ Presentation/ Group Discussion/ etc
ISEII: Internal Oral

Course Contents:

Experiment No. 1: Software Verification Methods **Aim and Objectives:** To create verification documents for various work products. **Theoretical Background:** Verification Methods**,** Review checklists. **Experimentation:** 1) Review SRS /Design Document	**02 Hrs.**
Experiment No. 2: Use Case Diagrams and Use Cases **Aim and Objectives:** To study Use Case Diagrams and Use cases. **Theoretical Background:** UML specification for Use Case Diagram. **Experimentation:** 1) Learn Use Case concept 2) Study UML notation for Use Case Diagram.	**02Hrs.**
Experiment No. 3: Test Cases **Aim and Objectives:** To create test cases from SRS and Use cases	**02Hrs.**

Theoretical Background: SRS document, Use Case, Test case Templates **Experimentation:** 1) Studying SRS document/Use Case diagram 2)Create Test Case actions respective to SRS & Use Case	
Experiment No.4: Test Plan and Test Case for given System **Aim and Objectives:** Create Test Plan and Test Case for given System **Theoretical Background:** Test Plan , Test Case **Experimentation:** 1) Study the given system (e.g. ATM/Library Management System) 2) Analyze Test case Actions, Expected results 3) Plan Test cases with respect to results in given system. 4) Design Test Plan and Test Cases.	**02Hrs.**
Experiment No. 5: Object Oriented Testing **Aim and Objectives:** To study Object Oriented Testing for given system. **Theoretical Background:** Object Oriented Testing method **Experimentation:** 1) Analyze Object Oriented aspect of given system	**02Hrs.**
Experiment No. 6: **Testing Tools** **Aim and Objectives:** To study various Testing Tools. **Theoretical Background:** Testing Techniques **Experimentation:** 1) Study Black Box Testing. 2) Study White Box Testing.	**02Hrs.**
Experiment No. 7: Manual Testing and Automation Testing **Aim and Objectives:** Compare between Manual Testing and Automation Testing **Theoretical Background:** Testing Methodologies **Experimentation:** 1.To study Manual Testing of any software system. 2. To demonstrate Automation Testing Tool.	**02Hrs.**
Experiment No. 8: Web Testing Tool. **Aim and Objectives:** To study the Web Testing Tool. **Theoretical Background:** Functional Testing. **Experimentation: 1)** To study Functional Testing.	**02Hrs.**
Experiment No. 9: Test Data Generation Tool. **Aim and Objectives:** To study Test Data Generation Tool. **Theoretical Background:** Test Units, Test Data Generation. **Experimentation:** 1) Generate Test data with respect to test case and SRS.	**02Hrs.**
Experiment No. 10: S/W Quality Modelling and Documentation. **Aim and Objectives:** To study Modelling S/W Quality and Documentation**.** **Theoretical Background:** Aspects of Software Quality **Experimentation: 1)** Study Software Quality attributes. **2)** Study aspects of Software Quality.	**02Hrs.**
Textbooks: 1. Software testing: Yogesh Singh, Cambridge University Press, First Edition 2. Software Metrics – A Rigorous & Practical approach: Norman Fenton, Shari Lawrence Pfleeger, 2nd Edition (Thomson Press) (unit 4 ,unit 5) 3. Software Quality Engineering, Jeff Tian, Wiley India Ltd.	
References: 1. Foundations of Software testing: Aditya P. Mathur, Pearson, Second Edition 2. Software Testing: Ron Patton, Pearson (SAMS), Second Edition 3. Software Quality, Mordechai Ben Menachem, Garry S. Marliss, BS Publications	

Title of the Course: Data Analytics Lab **Course Code:UCSE0634**	**L**	**T**	**P**	**Credit**
	--	--	2	1

Course Pre-Requisite: data warehousing, basics of mining techniques

Course Description:

Course Learning Objectives:

1. To understand Fundamental Concepts related to data analysis
2. To gain familiarity with R and Python language
3. To implement machine learning and data mining techniques for data analysis.

Course Outcomes:

CO	**After the completion of the course the student should be able to**
CO1	Apply different library functions in R and Python to solve data analytics problems
CO2	Implement machine learning and data mining algorithms to solve data analytics problem
CO3	Analysis and visualization of results

CO-PO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PO 12	PSO 1	PSO 2
CO1		2	1		2									2
CO2		2	1										1	
CO3		2			3								1	

Assessments :

Teacher Assessment:

One component of In Semester Evaluation (ISE), and one End Semester Examination (ESE) having 33% and 67% weights respectively.

Assessment	Marks
ISE	25
ESE	50

ISE assessment are based on practical performed/ Quiz/ Mini-Project assigned/ Presentation/ Group Discussion/ Internal oral etc.

ESE assessment is based on oral examination

Course Contents:

Experiment No. 1: Installation of R studio and executing basic R functions Aim and Objectives: To know installation procedure of R tool Outcomes: Students will able to know installation of R and execution of different commands Theoretical Background: The package provides various statistical methods for	**Hrs.**

designing and analyzing randomized experiments. Experimentation: 1. Install R studio andStudy of R-declaring variables, expressions, functions and executing R script. 2. Working with R with data frames- create, read, write and R Tables- create, read, and write. 3. Manipulating and processing data in R- merging datasets, sorting data, putting data into shape, managing data using matrices managing data using data frames. Conclusion: The package also provides the tools to analyze various randomized experiments including cluster randomized experiments, randomized experiments with noncompliance, and randomized experiments with missing data	
Experiment No. 2: Performing exploratory data analysis using Python Jupyter Editor Aim and Objectives: To pre-process data in data frames. Outcomes: Students will able to know data preprocessing techniques. Theoretical Background: The package provides various functions for exploratory data analysis. Experimentation: 1. Importing data into data frames 2. Apply different normalization techniques	**Hrs.**
Experiment No. 3: Exploring Data Analysis Libraries in Python Aim and Objectives: To detect outliers in the data within the data frames Outcomes: Students will able to outlier detection techniques. Theoretical Background: The package provides various functions for outlier detection Experimentation: 1. Importing data into data frames 2. Apply Numpy,Pandas to study List,Tuple,and Sets, Dictionaries Data Structure	**Hrs.**
Experiment No. 4: linear regression using Python Aim and Objectives: To use linear regression for finding relationship between depende and independent attributes Outcomes: Students will able to apply linear regression for finding relationship betwee dependent and independent attributes Theoretical Background: The package provides various functions for outlier detection	**Hrs.**

Experimentation: 1. Importing data into data frames 2. Draw scatter plot 3. Derive regression coefficients 4. Draw regression line 5. Identify relation	
Experiment No. 5: Analyze data distribution using graphs using matplotlib Aim and Objectives: To use graphs to analyse distribution of values. Outcomes: Students will able to analyse the distribution of data values for an attribute Theoretical Background: The package provides various functions for drawing differen types of graphs Experimentation: 1. Importing data into data frames 2. Draw bar chart, histogram etc 3. Analyze if the distribution of values is normal	**Hrs.**
Experiment No. 6: Create mathematical models using various Machine Learning Algo Aim and Objectives: To Create mathematical models using various Machine Learning in python for data analysis. Outcomes: Students will able to use basic libraries in python for data analysis.. Theoretical Background: The library provides various functions for data analysis 1. Libraries: NumPy, SciPy, Matplotlib, Pandas, Statsmodel etc.	**Hrs.**
Experiment No. 7: Preprocessing data using python Aim and Objectives: To use basic libraries in python for pre-processing data. Outcomes: Students will able to use basic libraries in python for data preprocessing Theoretical Background: The library provides various functions for data preprocessing Experimentation: 1. Apply different normalization techniques 2. Apply outlier detection techniques	
Experiment No. 8: Visualize data using python using Matplotlib Aim and Objectives: To use basic libraries in python for visualizing data.	

Outcomes: Students will able to use basic libraries in python for data visualization Theoretical Background: The library provides various functions for data visualization Experimentation: Draw scatter plots, histograms etc. to analyze distribution of values of the attribute	
Experiment No. 9: Visualize the data using Tableau Aim and Objectives: To use Tableau desktop version Outcomes: Students will able to visualize data using Tableau software Theoretical Background: Experimentation: Install Tableau desktop edition and display various data in the form of various charts like scatter plot, box plot, gnat chart etc.	
Textbooks: 1. VigneshPrajapati, Big data analytics with R and Hadoop, SPD 2013	
References: 1. https://www.tutorialspoint.com/tableau/tableau_environment_setup.htm	

Title of the Course: Adhoc wireless Network Lab **Course Code:** UCSE0635	L	T	P	Credit
	0	0	2	1

Course Pre-Requisite: Wireshark, computer network QoS parameters and basics of wireless network

Course Description: This course is designed to understand and analyse adhoc wireless network architecture and it protocols.

Course Objectives:

1. To understand architecture of ad hoc wireless architectures
2. To gain knowledge of wireless network simulators
3. To analyse routing protocols in ad hoc wireless network
4. To analyse energy management in ad hoc wireless network

Course Learning Outcomes:

CO	After the completion of the course the student should be able to
CO1	Understand & use simulation tools
CO2	Design wireless network scenarios and test it in simulator.
CO3	Design scenarios & evaluate different routing protocols
CO4	Understand energy/security parameters in wireless ad hoc networks

CO-PO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2
CO1					2								1	
CO2			2		2								1	
CO3			2		2								1	1
CO4					2								1	1

Assessments :

Teacher Assessment:

One component of In Semester Evaluation (ISE) and one End Semester Examination (ESE) having 50%, and 50% weights respectively.

Assessment	Marks
ISE	50
ESE	50

ISE are based on practical performed/ Quiz/ Mini-Project assigned/ Presentation/ Group Discussion/ Internal oral etc.

ESE: Assessment is based on oral examination

Course Contents:

Experiment No. 1: Study and Installation of NS2 **Aim and Objectives:** understand and install network simulator ns2 **Experimentation:** installation of ns2 on linux platform	**02 Hrs.**
Experiment No. 2: Study and installation of ns3 **Aim and Objectives:** .understand and install ns3 .	**02Hrs.**

Experimentation: installation of ns3 linux platform.	
Experiment No. 3: Study and install of OPNET **Aim and Objectives:** understand and install network simulator OPNET. **Experimentation:** installation of OPNET	**02Hrs.**
Experiment No. 4: Study and analyse pro-active routing protocols in ns2 **Aim and Objectives:** Analyse DSDV protocol in wireless ad hoc network**.** **Experimentation:** Design wireless network with DSDV routing protocol and trace packet transmission.	**02Hrs.**
Experiment No. 5: Study and analyse reactive routing protocol in ns2 **Aim and Objectives:** Analyse AODV protocol in wireless adhoc network**.** **Experimentation:** Design wireless network with AODV routing protocol and trace packet transmission.	**02Hrs.**
Experiment No. 6: Study and analyse ad hoc wireless network topology in ns3 **Aim and Objectives:** Analyse adhoc wireless network topology in ns3**.** **Experimentation:** Design ad hoc wireles network toplogy with UDP client and trace the packets	**02Hrs.**
Experiment No. 7: Study and analyse MAC protocol in adhoc wireless network **Aim and Objectives:** Analyse MAC protocol in ad hoc wireless network with respect throughput and delay. **Experimentation:** Design ad hoc wireless network topology with MAC protocol and trace the packets with respect to throughput and delay.	**02Hrs.**
Experiment No. 8: Study and analyse real time MANET in OPNET **Aim and Objectives:** Analyse real time MANET topology in OPNET**.** **Experimentation:** Design real time IEEE 802.11 network connected devices and analyse packet tracing.	**02Hrs.**
Experiment No. 9: Study and analyse multicast routing in ad hoc wireless network **Aim and Objectives:** Analyse multicast protocol in ad hoc wireless network with respect throughput and delay. **Experimentation:** Design ad hoc wireless network topology with multicast routing and trace the packets with respect to throughput and delay.	**02Hrs.**
Experiment No. 10: Study and analyse energy efficiency in ad hoc wireless network **Aim and Objectives:** Analyse energy efficiency of node in ad hoc wireless network **Experimentation:** Design ad hoc wireless network topology and analyse energy efficiency of selected node.	**02 Hrs.**

Title of the Course: UnixOperating System Lab	**L**	**T**	**P**	**Credit**
Course Code:UCSE0632	**0**	**0**	2	1

Course Pre-Requisite: Unix Operating System

Course Description: This course provides hands-on experience on Unix Features through installation, use of different system calls and basic building blocks provided by Unix.

Course Objectives:

1. To give competency with Unix and Linux based systems.
2. Understand working of different system calls.
3. To provide hands-on experience on Unix tools.

Course Learning Outcomes:

CO	**After the completion of the course the student should be able to**
CO1	Make use of system calls and tools provided by Unix and Linux based systems
CO2	Develop Shell Scripts.
CO3	Analyse various algorithms of Operating systems

CO-PO Mapping:

CO	**PO 1**	**PO 2**	**PO 3**	**PO 4**	**PO 5**	**PO 6**	**PO 7**	**PO 8**	**PO 9**	**PO 10**	**PO 11**	**PO 12**	**PS O1**	**PS O2**
CO1		2											1	
CO2				3									1	
CO3			2											

Assessments :
Teacher Assessment:

One component of In Semester Evaluation (ISE) and one End Semester Examination (ESE) having 50%, and 50% weights respectively.

Assessment	Marks
ISE	25
ESE(POE)	50

ISE are based on practical performed/ Quiz/ Internal oral etc.
ESE: Assessment is based on oral examination

Course Contents:	
Experiment No. 1:---Study & Installation of Unix Operating System	**2 Hrs.**
Experiment No. 2:---Command line navigation in Unix	**2 Hrs.**
Experiment No. 3:---Getting started with the shell programming	**2 Hrs.**
Experiment No. 4:---Implementation of buffer retrival algorithm	**2 Hrs.**
Experiment No. 5:--- Reading and Writing through Named and Unnamed Pipe	**2 Hrs.**
Experiment No. 6:---Study & demonstration of Unix Process Management	**2 Hrs.**
Experiment No. 7:--Study & demonstration of IPC mechanisms	**2 Hrs.**
Experiment No. 8:-- Study & demonstration of signal handler	**2 Hrs.**
Experiment No. 9:-- Study & demonstration of Time, Sleep and Clock Management	**2 Hrs.**
Experiment No. 10:-- Implementing Loadable Kernel Module for Linux	**2 Hrs.**
Textbooks: 1. The design of Unix Operating System - Maurice J. Bach (PHI)	
References: 1. Linux System Programming - Robert Love, Publisher - SPD, O' REILLY 2. Unix concepts and administration – 3rd Edition – Sumitabha Das (TMGH). 3. Unix / Linux Manuals.	
Subject In-charge Subject Expert Subject Expert	

Title of the Course: Project Development using advanced Tools Course Code: UCSE0651	L	T	P	Credits
	2	---	2	2

Course Pre-Requisite:Object Oriented Programming, Java and Database.

Course Objectives:

1. To describeandroid architecture and the tools for developing android Applications.
2. To create an android application.
3. To design the user interfaces used in android applications
4. To deploy android application on app market

Course Outcomes:

COs	After the completion of the course the student will be able to
CO1	Describe android architecture and the tools for developing android Applications.
CO2	Write program and create an android application.
CO3	Design user interfaces used in android applications.
CO4	Deploy their android applications.

CO-PO Mapping:

CO	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12	PSO 1	PSO 2
CO1	3				3									
CO2		3	3	2									2	
CO3		3	2	3	3									2
CO4						3		3			2	1		

Assessments :

Teacher Assessment:

Two components of In Semester Evaluation (ISE), One Mid Semester Examination (MSE) and one End Semester Examination (ESE) having 20%, 30% and 50% weights respectively.

Assessment	Marks
ISE	25
ESE (POE)	50

ISE will be based on assignment/declared test/quiz/seminar/Group Discussions etc.
ESE: Assessment is based on 100% course content with60-70% weight course content.

Course Contents:	
Unit 1: Android Overview Overview of Android, History, Android Versions, Android OS stack: Linux kernel, Native Libraries/DVM, Application Framework, Applications,Activity, Activity lifecycle, Fragments, Activity Back Stack, Process and Threads.	**3Hrs.**
Unit 2: Android Development Environment Introduction to Android SDK, Android Emulator, Creating a Project, Project Directory Structure, Logging in Android (Logcat), Managing Exception with Logcat, Android Manifest File, Android Resources and Resource Directories, Permissions.	**3Hrs.**
Unit 3: Intents and Layouts XML, Android View Hierarchies, Linear Layouts, Relative Layout, Table Layout, Frame Layout, Padding and Margins with Layouts.What Is Intent? Implicit & Explicit Intents, Android Intent Messaging via Intent Objects, Using Intentswith Activities, Sending Intents (Telephony, SMS), Broadcast Receivers.	**5Hrs.**
Unit 4: Input Controls, Input Events, Dialogs Buttons, Text Fields, Checkboxes, Radio Buttons, Toggle Buttons, Custom List, Grids,Spinners, Event Listeners, AsyncTasks.Event Handlers, Touch Mode, Handling Focus, Dialogs: Alerts, Custom Dialogs, Toasts.	**7Hrs.**
Unit 5: Menus, Notification and Action Bar Menus, Options menu, Context menu, Popup menu, Handling menu click events, Creating a Notification, Notification actions, Notification priority, Managing Notifications, Removing notifications.	**5Hrs.**
Unit 6: Android SQLiteand App Market Installing SQLite plugin, DbHelper, The Database Schema and Its Creation, Four MajorOperations, Adding External Libraries to android.	**5Hrs.**

Text Books:

1. Beginning Android application development by Wei-Mag Lee.
2. Learning Android by Marko Gargenta Publisher: O'Reilly Media

Reference Books:

1. Professional Android 4 Application Development by Reto Meier Publisher: Wiley India.
2. Android in Action Third Edition by W.Frank Ableson,Robi Sen, Chris King, C. Enrique Ortiz.
3. The Android Developer’s Cook book “Building Applications with the Android SDK” by James Steele.

List of Experiments:	**Hours**
Experiment No.1: Installation and Configuration of Android App Development Environment.	**02 Hrs.**
Experiment No.2: Study of Android Activity lifecycle	**02 Hrs.**
Experiment No.3: Design of Android User interfaces using XML layouts	**02 Hrs.**
Experiment No.4: Android explicit Intents	**02 Hrs.**
Experiment No.5: Android implicit Intents	**02 Hrs.**
Experiment No.6: Implementation of User login and registration using button Event handling	**02 Hrs.**
Experiment No.7: Event Handling for other controls with asyn	**02 Hrs.**
Experiment No.8: Creating broadcast receiver in androids and responding to broadcast messages	**02 Hrs.**
Experiment No.9: Creating Notifications in android	**02 Hrs.**
Experiment No.10: Writing custom broadcast messages and receiving them	**02 Hrs.**
Experiment No.11: Experiment based on database handling using SQLite through android form.(Insert, Update, Delete Records)	**02 Hrs.**

Title of the Course: Professional Communication	**L**	**T**	**P**	**Credit**
Course Code: UCSE0661	--	--	2	--

Course Pre-Requisite: Soft Skill

Course Description: Understanding and developing skills in communication are fundamental to all levels of human interaction. For successful participation in professional and social life, students must develop effective communication skills. Rapidly expanding technologies and changing social and corporate systems demand that students send clear verbal messages, choose effective nonverbal behaviors, listen for desired results, and apply valid critical-thinking and problem-solving processes. Students enrolled in Professional Communication will be expected to identify, analyze, develop, and evaluate communication skills needed for professional and social success in interpersonal situations, group interactions, and personal and professional presentations.

Course Learning Objectives:

1. To describe students the purpose and goal of professional communication.
2. To understand effective communication components.
3. To discuss characteristics of effective communication.
4. To describe Etiquette's contribution to professional communication.

Course Outcomes:

CO	**After the completion of the course the student should be able to**
CO1	Understand the role of communication in personal & professional success.
CO2	Analyze a variety of communication acts based in team activities
CO3	Develop awareness of appropriate communication strategies
CO4	Prepare corporate writing skills with a specific intent

CO-PO Mapping:

CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO 10	PO 11	PO 12	PSO1	PSO2
CO1										3				
CO2									3		1			
CO3						3								
CO4					2						1			2

Assessments :

Teacher Assessment:

End Semester Examination (ESE) having 100% weight.

Course Contents:

Unit – I Basics of Communication skill: Introduction to English grammar, Vocabulary, types of communication-Verbal and Non Verbal	**3 Hrs.**
Unit – II Importance of working in a TEAM:	**4 Hrs.**

Concept of Team, Efficient team, characteristic of Team leader, Team based activities-Group Discussions, Debate, Decision making.	
Unit - III Technical Communication Forms: Business Letters: Letter of Enquiry; Sales and Credit letters; Letter of Quotation, Order, Claim and Adjustment Letters; Job application and Resumes. Official Letters: D.O. Letters; Govt. Letters, Letters to Authorities etc.	**4 Hrs.**
Unit – IV: Reports: Types; Significance; Structure, Style & Writing of Reports. Technical Proposal; Parts; Types; Writing of Proposal; Significance. Technical Paper, Project. Dissertation and Thesis Writing: Features, Methods & Writing.	**4 Hrs.**
Unit - V Public Speaking and Presentation Strategies: Overcoming stage fear ; Defining Purpose; Audience & Locale; Organizing Contents; Preparing Outline; Audio-visual Aids; Nuances of Delivery; Body Language; Space; Setting Nuances of Voice Dynamics; Time- Dimension.	**5 Hrs.**
Unit - VI: Miscellaneous: Issues Escalation- Handling complaints, Practice of Right to Information (RTI), Business ethics, ethics Policy, business etiquette, dining etiquette, Negotiations.	**4 Hrs.**
Textbooks: 1. Improve Your Writing ed. V.N. Arora and Laxmi Chandra, Oxford Univ. Press, New Delhi Technical Communication – Principles and Practices by Meenakshi Raman & Sangeeta Sharma, Oxford Univ. Press 2007, New Delhi.	
References: 1. Effective Technical Communication by Barun K. Mitra, Oxford Univ. Press, 2006, New Delhi Business Correspondence and Report Writing by Prof. R.C. Sharma & Krishna Mohan, Tata McGraw Hill&Co.Ltd.,NewDelhi.	