

## IV YEAR I SEMESTER

## COURSE STRUCTURE

Code	Subject	L	T/P/D	C
57046	Linux Programming	4	1	4
57047	Software Testing Methodologies	4	1	4
57048	Data Warehousing and Data Mining	4	–	4
57049	Computer Graphics	3	1	3
	<b>ELECTIVE-I</b>	3	1	3
57050	Advanced Computer Architecture			
57051	Cloud Computing			
57052	Distributed Computing			
57053	Mobile Computing			
	<b>ELECTIVE-II</b>	3	1	3
57054	Design Patterns			
57055	Machine Learning			
57056	Soft Computing			
57057	Information Retrieval Systems			
57609	Linux Programming and Data Mining Lab	–	3	2
57610	Case Tools & Software Testing Lab	–	3	2
	<b>Total</b>	<b>21</b>	<b>11</b>	<b>25</b>

**JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY  
HYDERABAD**

<b>IV Year B.Tech. CSE - I Sem</b>	<b>L</b>	<b>T/P/D</b>	<b>C</b>
	4	1/-/-	4

**(57046) LINUX PROGRAMMING**

**UNIT - I**

Linux Utilities-File handling utilities, Security by file permissions, Process utilities, Disk utilities, Networking commands, Filters, Text processing utilities and Backup utilities, sed – scripts, operation, addresses, commands, applications, awk – execution, fields and records, scripts, operation, patterns, actions, functions, using system commands in awk.

**UNIT-II**

Working with the Bourne again shell(bash): Introduction, shell responsibilities, pipes and input Redirection, output redirection, here documents, running a shell script, the shell as a programming language, shell meta characters, file name substitution, shell variables, command substitution, shell commands, the environment, quoting, test command, control structures, arithmetic in shell, shell script examples, interrupt processing, functions, debugging shell scripts.

**UNIT - III**

Files: File Concept, File System Structure, Inodes, File Attributes, File types, Library functions, the standard I/O and formatted I/O in C, stream errors, kernel support for files, System calls, file descriptors, low level file access – File structure related system calls(File APIs), file and record locking, file and directory management – Directory file APIs, Symbolic links & hard links.

**UNIT - IV**

Process – Process concept, Kernel support for process, process attributes, process control - process creation, waiting for a process, process termination, zombie process, orphan process, Process APIs.  
Signals– Introduction to signals, Signal generation and handling, Kernel support for signals, Signal function, unreliable signals, reliable signals, kill, raise, alarm, pause, abort, sleep functions.

**UNIT - V**

Interprocess Communication : Introduction to IPC, Pipes, FIFOs, Introduction to three types of IPC-message queues, semaphores and shared memory.

Message Queues- Kernel support for messages, Unix system V APIs for messages, client/server example.

#### UNIT-VI

Semaphores-Kernel support for semaphores, Unix system V APIs for semaphores.

Shared Memory- Kernel support for shared memory, Unix system V APIs for shared memory, semaphore and shared memory example.

#### UNIT-VII

Multithreaded Programming: Differences between threads and processes, Thread structure and uses, Threads and Lightweight Processes, POSIX Thread APIs, Creating Threads, Thread Attributes, Thread Synchronization with semaphores and with Mutexes, Example programs.

#### UNIT-VIII

Sockets: Introduction to Sockets, Socket Addresses, Socket system calls for connection oriented protocol and connectionless protocol, example-client/server programs.

#### TEXT BOOKS:

1. Unix System Programming using C++, T.Chan, PHI.(UNIT III to UNIT VIII)
2. Unix Concepts and Applications, 4th Edition, Sumitabha Das, TMH.
3. Beginning Linux Programming, 4<sup>th</sup> Edition, N.Matthew, R.Stones,Wrox, Wiley India Edition.

#### REFERENCE BOOKS:

1. Linux System Programming, Robert Love, O'Reilly, SPD.
2. Advanced Programming in the Unix environment, 2<sup>nd</sup> Edition, W.R.Stevens, Pearson Education.
3. Unix Network Programming, W.R.Stevens,PHI.
4. Unix for programmers and users, 3<sup>rd</sup> Edition, Graham Glass, King Ables, Pearson Education.
5. Unix and Shell programming, B.A.Forouzan and R.F.Gilberg, Cengage Learning.
6. Unix The Text book, 2<sup>nd</sup> edition, S.M.Sarwar, R.Koretsky, S.A.Sarwar, Pearson Education.
7. Unix Internals, U.Vahalia, Pearson Education.
8. Unix shell Programming, S.G.Kochan and P.Wood, 3<sup>rd</sup> edition, Pearson Education.

## JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY

### HYDERABAD

IV Year B.Tech. CSE - I Sem

L	T/P/D	C
4	1/-/	4

### (57047) SOFTWARE TESTING METHODOLOGIES

#### UNIT-I

Introduction:- Purpose of testing, Dichotomies, model for testing, consequences of bugs, taxonomy of bugs.

#### UNIT-II

Flow graphs and Path testing:- Basics concepts of path testing, predicates, path predicates and achievable paths, path sensitizing, path instrumentation, application of path testing.

#### UNIT-III

Transaction Flow Testing:-transaction flows, transaction flow testing techniques. Dataflow testing:- Basics of dataflow testing, strategies in dataflow testing, application of dataflow testing.

#### UNIT-IV

Domain Testing:-domains and paths, Nice & ugly domains, domain testing, domains and interfaces testing, domain and interface testing, domains and testability.

#### UNIT-V

Paths, Path products and Regular expressions:- path products & path expression, reduction procedure, applications, regular expressions & flow anomaly detection.

#### UNIT-VI

Logic Based Testing:- overview, decision tables, path expressions, kv charts, specifications.

#### UNIT-VII

State, State Graphs and Transition testing:- state graphs, good & bad state graphs, state testing, Testability tips.



**UNIT - VIII**

Graph Matrices and Application:-Motivational overview, matrix of graph, relations, power of a matrix, node reduction algorithm, building tools. ( Student should be given an exposure to a tool like JMeter or Win-runner).

**TEXT BOOKS:**

1. Software Testing techniques – Boris Beizer, Dreamtech, second edition.
2. Software Testing Tools – Dr.K.V.K.K.Prasad, Dreamtech.

**REFERENCE BOOKS:**

1. The craft of software testing - Brian Marick, Pearson Education.
2. Software Testing, 3<sup>rd</sup> edition, P.C.Jorgensen, Aurbach Publications (Dist.by SPD).
3. Software Testing, N.Chauhan, Oxford University Press.
4. Introduction to Software Testing, P.Ammann & J.Offutt, Cambridge Univ.Press.
5. Effective methods of Software Testing, Perry, John Wiley, 2<sup>nd</sup> Edition, 1999.
6. Software Testing Concepts and Tools, P.Nageswara Rao, dreamtech Press.
7. Software Testing, M.G.Limaye, TMH.
8. Software Testing, S.Desikan, G.Ramesh, Pearson.
9. Foundations of Software Testing, D.Graham & Others, Cengage Learning.
10. Foundations of Software Testing, A.P.Mathur, Pearson.

**JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY  
HYDERABAD**

IV Year B.Tech. CSE - I Sem	<b>L</b>	<b>T/P/D</b>	<b>C</b>
	<b>4</b>	<b>-/-</b>	<b>4</b>

**(57048) DATA WAREHOUSING AND DATA MINING**

**UNIT I**

Introduction: Fundamentals of data mining, Data Mining Functionalities, Classification of Data Mining systems, Data Mining Task Primitives, Integration of a Data Mining System with a Database or a Data Warehouse System, Major issues in Data Mining.  
Data Preprocessing: Need for Preprocessing the Data, Data Cleaning, Data Integration and Transformation, Data Reduction, Discretization and Concept Hierarchy Generation.

**UNIT II**

Data Warehouse and OLAP Technology for Data Mining: Data Warehouse, Multidimensional Data Model, Data Warehouse Architecture, Data Warehouse Implementation, Further Development of Data Cube Technology, From Data Warehousing to Data Mining  
Data Cube Computation and Data Generalization: Efficient Methods for Data Cube Computation, Further Development of Data Cube and OLAP Technology, Attribute-Oriented Induction.

**UNIT III**

Mining Frequent Patterns, Associations and Correlations: Basic Concepts, Efficient and Scalable Frequent Itemset Mining Methods, Mining various kinds of Association Rules, From Association Mining to Correlation Analysis, Constraint-Based Association Mining

**UNIT IV**

Classification and Prediction: Issues Regarding Classification and Prediction, Classification by Decision Tree Induction, Bayesian Classification, Rule-Based Classification, Classification by Backpropagation, Support Vector Machines, Associative Classification, Lazy Learners, Other Classification Methods, Prediction, Accuracy and Error measures, Evaluating the accuracy of a Classifier or a Predictor, Ensemble Methods

**Unit V**

Cluster Analysis Introduction :Types of Data in Cluster Analysis, A Categorization of Major Clustering Methods, Partitioning Methods, Hierarchical Methods, Density-Based Methods, Grid-Based Methods, Model-Based Clustering Methods, Clustering High-Dimensional Data, Constraint-Based Cluster Analysis, Outlier Analysis.

**UNIT VI**

Mining Streams, Time Series and Sequence Data: Mining Data Streams, Mining Time-Series Data, Mining Sequence Patterns in Transactional Databases, Mining Sequence Patterns in Biological Data, Graph Mining, Social Network Analysis and Multirelational Data Mining:

**UNIT VII**

Mining Object, Spatial, Multimedia, Text and Web Data: Multidimensional Analysis and Descriptive Mining of Complex Data Objects, Spatial Data Mining, Multimedia Data Mining, Text Mining, Mining the World Wide Web.

**UNIT VIII**

Applications and Trends in Data Mining: Data Mining Applications, Data Mining System Products and Research Prototypes, Additional Themes on Data Mining and Social Impacts of Data Mining.

**TEXT BOOKS:**

1. Data Mining – Concepts and Techniques - Jiawei Han & Micheline Kamber, Morgan Kaufmann Publishers, Elsevier, 2<sup>nd</sup> Edition, 2006.
2. Introduction to Data Mining – Pang-Ning Tan, Michael Steinbach and Vipin Kumar, Pearson education.

**REFERENCE BOOKS:**

1. Data Mining Techniques – Arun K Pujari, 2<sup>nd</sup> edition, Universities Press.
2. Data Warehousing in the Real World – Sam Aanhory & Dennis Murray Pearson Edn Asia.
3. Insight into Data Mining, K.P.Soman, S.Diwakar, V.Ajay, PHI, 2008.
4. Data Warehousing Fundamentals – Paulraj Ponnaiah Wiley student Edition
5. The Data Warehouse Life cycle Tool kit – Ralph Kimball Wiley student edition
6. Building the Data Warehouse By William H Inmon, John Wiley & Sons Inc. 2005.
7. Data Mining Introductory and advanced topics – Margaret H Dunham, Pearson education
8. Data Mining, V.Pudi and P.Radha Krishna, Oxford University Press.
9. Data Mining: Methods and Techniques, A.B.M Shawkat Ali and S.A.Wasimi, Cengage Learning.
10. Data Warehouse 2.0, The Architecture for the next generation of Data Warehousing, W.H.Inmon, D.Strauss, G.Neushloss, Elsevier, Distributed by SPD.

**JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY  
HYDERABAD**

<b>IV Year B.Tech. CSE - I Sem</b>	<b>L</b>	<b>T/P/D</b>	<b>C</b>
	3	1/-/-	3

**(57049) COMPUTER GRAPHICS**

**UNIT I**

Introduction, Application areas of Computer Graphics, overview of graphics systems, video-display devices, raster-scan systems, random scan systems, graphics monitors and work stations and input devices

**UNIT II**

Output primitives: Points and lines, line drawing algorithms, mid-point circle and ellipse algorithms. Filled area primitives: Scan line polygon fill algorithm, boundary-fill and flood-fill algorithms.

**UNIT III**

2-D Geometrical transforms: Translation, scaling, rotation, reflection and shear transformations, matrix representations and homogeneous coordinates, composite transforms, transformations between coordinate systems.

**UNIT IV**

2-D Viewing : The viewing pipeline, viewing coordinate reference frame, window to view-port coordinate transformation, viewing functions, Cohen-Sutherland and Cyrus-beck line clipping algorithms, Sutherland –Hodgeman polygon clipping algorithm.

**UNIT V**

3-D Object representation: Polygon surfaces, quadric surfaces, spline representation, Hermite curve, Bezier curve and B-spline curves, Bezier and B-spline surfaces. Basic illumination models, polygon rendering methods.

**UNIT VI**

3-D Geometric transformations: Translation, rotation, scaling, reflection and



shear transformations, composite transformations, 3-D viewing: Viewing pipeline, viewing coordinates, view volume and general projection transforms and clipping.

#### UNIT VII

Visible surface detection methods: Classification, back-face detection, depth-buffer, scan-line, depth sorting, BSP-tree methods, area sub-division and octree methods

#### UNIT VIII

Computer animation: Design of animation sequence, general computer animation functions, raster animation, computer animation languages, key frame systems, motion specifications

#### TEXT BOOKS:

1. "Computer Graphics C version", Donald Hearn and M. Pauline Baker, Pearson education.
2. "Computer Graphics Principles & practice", second edition in C, Foley, VanDam, Feiner and Hughes, Pearson Education.

#### REFERENCE BOOKS:

1. "Computer Graphics Second edition", Zhigang xiang, Roy Plastock, Schaum's outlines, Tata Mc Graw hill edition.
2. "Procedural elements for Computer Graphics", David F Rogers, Tata Mc Graw hill, 2nd edition.
3. "Principles of Interactive Computer Graphics", Neuman and Sproul, TMH.
4. "Principles of Computer Graphics", Shalini, Govil-Pai, Springer.
5. "Computer Graphics", Steven Harrington, TMH
6. Computer Graphics, F.S.Hill, S.M.Kelley, PHI.
7. Computer Graphics, P.Shirley, Steve Marschner & Others, Cengage Learning.
8. Computer Graphics & Animation, M.C.Trivedi, Jaico Publishing House.
9. An Integrated Introduction to Computer Graphics and Geometric Modelling, R.Goldman, CRC Press, Taylor & Francis Group.
10. Computer Graphics, Rajesh K. Maurya, Wiley India.

## JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY

### HYDERABAD

IV Year B.Tech. CSE - I Sem	L	T/P/D	C
	3	1/-	3

#### (57050) ADVANCED COMPUTER ARCHITECTURE (ELECTIVE-I)

##### Unit-I

Theory of Parallelism, Parallel computer models, The State of Computing, Multiprocessors and Multicomputers, Multivector and SIMD Computers, PRAM and VLSI models, Architectural development tracks, Program and network properties, Conditions of parallelism, Program partitioning and Scheduling, Program flow Mechanisms, System interconnect Architectures.

##### Unit-II

Principals of Scalable performance, Performance metrics and measures, Parallel Processing applications, Speed up performance laws, Scalability Analysis and Approaches, Hardware Technologies, Processes and Memory Hierarchy, Advanced Processor Technology, Superscalar and Vector Processors, Memory Hierarchy Technology, Virtual Memory Technology.

##### Unit-III

Bus Cache and Shared memory, Backplane bus systems, Cache Memory organizations, Shared-Memory Organizations, Sequential and weak consistency models, Pipelining and superscalar techniques, Linear Pipeline Processors, Non-Linear Pipeline Processors, Instruction Pipeline design, Arithmetic pipeline design, superscalar pipeline design.

##### Unit-IV

Parallel and Scalable Architectures, Multiprocessors and Multicomputers, Multiprocessor system interconnects, cache coherence and synchronization mechanism, Three Generations of Multicomputers, Message-passing Mechanisms, Multivector and SIMD computers, Vector Processing Principals, Multivector Multiprocessors, Compound Vector processing, SIMD computer Organizations, The connection machine CM-5,

##### Unit-V

Scalable, Multithreaded and Dataflow Architectures, Latency-hiding techniques, Principals of Multithreading, Fine-Grain Multicomputers, Scalable and multithreaded Architectures, Dataflow and hybrid

Architectures.

### Unit – VI

Software for parallel programming, Parallel models, Languages and Compilers, Parallel Programming models, Parallel languages and compilers, Dependence analysis and data arrays, code optimization and scheduling, Loop Parallelization and pipelining,

### Unit – VII

Parallel Program development and Environments, Parallel Programing Environments, Synchronization and Multiprocessing modes, Shared-Variable program structures, Message-passing program development. Mapping program onto multicomputers.

### Unit – VIII

Instruction level parallelism, Introduction, Basic Design issues, Problem Definition, Model of typical processor, Compiler-Detector Instruction level parallelism, Operand forwarding, Recorder Buffer, Register Re-naming, Tomasulo's Algorithm, Branch Prediction, Limitations in exploiting instruction level parallelism, Thread level parallelism, Recent Advances in computer Architecture, Brief overview of Technology, Forms of Parallelism.

### TEXT BOOK:

1. Advanced Computer Architecture Second Edition, Kai Hwang, Tata McGraw Hill Publishers.

### REFERENCE BOOKS:

1. Computer Architecture, Fourth edition, J.L.Hennessy and D.A. Patterson. ELSEVIER.
2. Advanced Computer Architectures, S.G.Shiva, Special Indian edition, CRC, Taylor & Francis.
3. Introduction to High Performance Computing for Scientists and Engineers, G.Hager and G.Wellein, CRC Press, Taylor & Francis Group.
4. Advanced Computer Architecture, D.Sima, T .Fountain, P.Kacsuk, Pearson education.
5. Computer Architecture, B.Parhami, Oxford Univ. Press.

## JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD

IV Year B.Tech. CSE - I Sem	L	T/P/D	C
	3	1/-	3

### (57051) CLOUD COMPUTING (ELECTIVE-I)

#### UNIT-I

Introductory concepts and overview: Distributed systems – Parallel computing architectures: Vector processing, Symmetric multi processing and Massively parallel processing systems – High performance Cluster computing – Grid computing – Service Oriented Architecture overview – Virtualization.

#### UNIT-II

Overview of Cloud Computing: Meaning of the terms cloud and cloud computing – cloud based service offerings – Grid computing Vs Cloud computing – Benefits of cloud model – limitations – legal issues – Key characteristics of cloud computing – Challenges for the cloud – The evolution of cloud computing.

#### UNIT-III

Web services delivered from the cloud: Infrastructure as a service – Platform-as-a-service – Software-as-a-service. Building Cloud networks: Evolution from the MSP model to cloud computing and software-as-a-service – The cloud data center – SOA as step toward cloud computing – Basic approach to a data center based SOA.

#### UNIT-IV

Federation Presence, Identity and Privacy in the cloud: Federation in the cloud – Presence in the cloud – Privacy and its relation to cloud based information system. Security in the Cloud: Cloud security challenges – Software-as-a-service security

#### UNIT-V

Common Standards in Cloud computing: The open cloud consortium – The



distributed management task force – standards for application developers  
 -- standards for messaging – standards for security

#### UNIT – VI

End user access to cloud computing: youtube – zimbra – Facebook – Zoho  
 – DimDim Collaboration

Mobile internet devices and the cloud: Smartphone – mobile operating systems for smart phones – Mobile Platform virtualization – Collaboration applications for mobile platforms – Future trends

#### UNIT – VII

Virtualization: Adding guest Operating system. Cloud computing case studies 1: Amazon EC2 – Amazon simple DB – Amazon S3 – Amazon Cloud Front – Amazon SQS

#### UNIT – VIII

Cloud computing case studies 2: Google App Engine- Google web tool kit – Microsoft Azure Services platform – Windows live – Exchange on line – Sharepoint services – Microsoft dynamic CRM – salesforce.com CRM – App Exchange

#### TEXT BOOKS:

1. Cloud Computing implementation, management and security by John W. Rittinghouse, James F. Ransome, CRC Press, Taylor & Francis group, 2010
2. Cloud Computing a practical approach by Anthony T. velte, Toby J. velte Robert Elsenpeter, Tata Mc Graw Hill edition, 2010

#### REFERENCES:

1. Cloud Application Architectures by George Reese, Oreilly publishers
2. Cloud computing and SOA convergence in your enterprise, by David S. Linthicum, Addison- Wesley

### JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY

#### HYDERABAD

IV Year B.Tech. CSE - I Sem

L	T/P/D	C
3	1/-/-	3

#### (57052) DISTRIBUTED COMPUTING

#### (ELECTIVE -I)

#### UNIT I

Introduction

Definitions, The different forms of computing – Monolithic, Distributed, Parallel and cooperative computing, the meaning of Distributed computing, Examples of Distributed systems, the strengths and weaknesses of Distributed computing, operating system concepts relevant to distributed computing, Network basics, the architecture of distributed applications, Interprocess Communications- An Archetypal IPC Program Interface, Event Synchronization, Timeouts and Threading, Deadlocks and Timeouts, Data representation,

Data Encoding, Text-Based Protocols, Request-Response Protocols, Event Diagram and Sequence Diagram, Connection- Oriented versus Connectionless IPC, The Evolution of Paradigms for IPCs.

#### UNIT II

Distributed Computing Paradigms

Paradigms and Abstraction, Paradigms for Distributed Applications – Message Passing Paradigm, The Client-Server Paradigm, The peer-to-peer Paradigm, Message system (or MOM) Paradigm – the point-to-point message model and the publish/subscribe message model, RPC model, The Distributed Objects Paradigms – RMI, ORB, the object space Paradigm, The Mobile Agent Paradigm, the Network Services Paradigm, The collaborative application ( Groupware Paradigm) ,choosing a Paradigm for an application.

#### UNIT III

The Socket API-The Datagram Socket API, The Stream-Mode Socket



API, Client-Server Paradigm Issues, Connection-Oriented and Connectionless Servers, Iterative and Concurrent Servers.

Group Communication-Unicasting versus Multicasting, Multicast API, Connectionless versus Connection-Oriented Multicast, Reliable Multicasting versus Unreliable Multicasting, The Java Basic Multicast API.

#### UNIT IV

Distributed Objects Paradigm (RMI)

Message passing versus Distributed Objects, An Archetypal Distributed Object Architecture, Distributed Object Systems, RPC, RMI, The Java RMI Architecture, Java RMI API, A sample RMI Application, steps for building an RMI application, testing and debugging, comparison of RMI and socket API.

#### UNIT V

Distributed Object Paradigm (CORBA)

The basic Architecture, The CORBA object interface, Inter-ORB protocols, object servers and object clients, CORBA object references, CORBA Naming Service and the Interoperable Naming Service, CORBA object services, object Adapters, Java IDL, An example CORBA application.

#### UNIT VI

Grid Computing

Introduction, Grid Computing Anatomy – The Grid Problem, The Concept of Virtual Organizations, Grid Architecture, Grid Architecture and relationship to other Distributed Technologies, Grid computing road map. Merging the Grid services Architecture with the Web Services Architecture.

#### UNIT VII

Open Grid Service Architecture – Introduction, Architecture and Goal, Sample Use cases: Commercial Data Center, National Fusion Collaboratory, Online Media and Entertainment. OGSA platform Components, Open Grid Services Infrastructure.

#### UNIT VIII

Globus GT 3 Toolkit – Architecture, Programming Model, A sample

implementation.

#### TEXT BOOKS:

1. Distributed Computing, Principles and Applications, M.L.Liu, Pearson Education.
2. Grid Computing, Joshy Joseph & Craig Fellenstein, Pearson education, 2004

#### REFERENCE BOOKS:

1. A Networking Approach to Grid Computing, D.Minoli, Wiley & sons.
2. Grid Computing: A Practical Guide to Technology and Applications, A.Abbas, Firewall Media.
3. Java Network Programming, E.R.Harold, 2nd edition, O'Reilly, SPD.
4. Distributed Systems, Concepts and Design, 3rd edition, G.Coulouris, J.Dollimore and Tim Kindbirg, Pearson Education.
5. Java Programming with CORBA, 3rd edition, Brose, Vogel, Duddy, Wiley Dreamtech.
6. Client/Server Programming with Java and CORBA, second edition, R.Orfali & Dan Harkey, John Wiley & sons.
7. Grid Computing – Making the global infrastructure a reality, Fran Berman, Geoffrey C Fox, Anthony J G Hey, Wiley India, 2010

**JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY  
HYDERABAD**

<b>IV Year B.Tech. CSE - I Sem</b>	<b>L</b>	<b>T/P/D</b>	<b>C</b>
	<b>3</b>	<b>1/-/-</b>	<b>3</b>

**(57053) MOBILE COMPUTING  
(ELECTIVE - I)**

**UNIT I**

## Introduction:

Mobile Communications, Mobile Computing – Paradigm, Promises/Novel Applications and Impediments and Architecture; Mobile and Handheld Devices, Limitations of Mobile and Handheld Devices.

GSM – Services, System Architecture, Radio Interfaces, Protocols, Localization, Calling, Handover, Security, New Data Services, GPRS, CSHSD, DECT.

**UNIT-II**

(Wireless) Medium Access Control (MAC)

Motivation for a specialized MAC (Hidden and exposed terminals, Near and far terminals), SDMA, FDMA, TDMA, CDMA. MAC protocols for GSM, Wireless LAN (IEEE802.11), Collision Avoidance (MACA, MACAW) Protocols.

**UNIT-III**

Mobile IP Network Layer

IP and Mobile IP Network Layers, Packet Delivery and Handover Management, Location Management, Registration, Tunnelling and Encapsulation, Route Optimization, DHCP.

**UNIT-IV**

Mobile Transport Layer

Conventional TCP/IP Protocols, Indirect TCP, Snooping TCP, Mobile TCP, Other Transport Layer Protocols for Mobile Networks.

**UNIT V**

Database Issues

Database Hoarding & Caching Techniques, C – S Computing & Adaptation, Transactional Models, Query processing, Data Recovery Process & QoS Issues.

**UNIT VI**

Data Dissemination and Synchronization, Communications Asymmetry, Classification of Data Delivery Mechanisms, Data Dissemination Broadcast Models, Selective Tuning and Indexing Methods, Digital Audio and Video Broadcasting (DAB & DVB). Data Synchronization – Introduction, Software, and Protocols

**UNIT VII**

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.

**UNIT – VIII**

Protocols and Platforms for Mobile Computing WAP, Bluetooth, XML, J2ME, JavaCard, PalmOS, Windows CE, SymbianOS, Linux for Mobile Devices.

**TEXT BOOKS**

1. Raj Kamal, "Mobile Computing", Oxford University Press, 2007, ISBN: 0195686772

**REFERENCE BOOKS**

1. Jochen Schiller, "Mobile Communications", Addison-Wesley, Second Edition, 2004.
2. Stojmenovic and Cacute, "Handbook of Wireless Networks and Mobile Computing", Wiley, 2002, ISBN 0471419028.
3. Reza Behravanfar, "Mobile Computing Principles: Designing and Developing Mobile Applications with UML and XML", ISBN: 0521817331, Cambridge University Press, Oct 2004,



## JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY

## HYDERABAD

IV Year B.Tech. CSE - I Sem	L	T/P/D	C
	3	1/-/-	3

**(57054) DESIGN PATTERNS****(ELECTIVE – II)****UNIT-I**

Introduction : What Is a Design Pattern?, Design Patterns in Smalltalk MVC, Describing Design Patterns, The Catalog of Design Patterns, Organizing the Catalog, How Design Patterns Solve Design Problems, How to Select a Design Pattern, How to Use a Design Pattern.

**UNIT-II**

A Case Study : Designing a Document Editor : Design Problems, Document Structure, Formatting, Embellishing the User Interface, Supporting Multiple Look-and-Feel Standards, Supporting Multiple Window Systems, User Operations Spelling Checking and Hyphenation, Summary .

**UNIT-III**

Creational Patterns : Abstract Factory, Builder, Factory Method, Prototype, Singleton, Discussion of Creational Patterns.

**UNIT-IV**

Structural Pattern Part-I : Adapter, Bridge, Composite.

**UNIT-V**

Structural Pattern Part-II : Decorator, açade, Flyweight, Proxy.

**UNIT-VI**

Behavioral Patterns Part-I : Chain of Responsibility, Command, Interpreter, Iterator.

**UNIT-VII**

Behavioral Patterns Part-II : Mediator, Memento, Observer, State, Strategy, Template Method , Visitor, Discussion of Behavioral Patterns.

**UNIT-VIII**

What to Expect from Design Patterns, A Brief History, The Pattern Community An Invitation, A Parting Thought.

**TEXT BOOK :**

1. Design Patterns By Erich Gamma, Pearson Education
2. Head First Design Patterns By Eric Freeman-Oreilly-SPD.

**REFERENCES:**

1. Pattern's in JAVA Vol-I By Mark Grand ,Wiley DreamTech.
2. Pattern's in JAVA Vol-II By Mark Grand ,Wiley DreamTech.
3. JAVA Enterprise Design Patterns Vol-III By Mark Grand ,Wiley DreamTech.
4. Design Patterns Explained By Alan Shalloway,Pearson Education.
5. Pattern Oriented Software Architecture,F.Buschmann&others,John Wiley & Sons.

**JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY  
HYDERABAD**

**IV Year B.Tech. CSE - I Sem**

<b>L</b>	<b>T/P/D</b>	<b>C</b>
3	1/-/-	3

**(57055) MACHINE LEARNING  
(ELECTIVE - II)**

**UNIT - I**

Introduction: An illustrative learning task, and a few approaches to it. What is known from algorithms? Theory, Experiment. Biology. Psychology.

**UNIT - II**

Concept Learning: Version spaces. Inductive Bias. Active queries. Mistake bound/ PAC model. basic results. Overview of issues regarding data sources, success criteria.

**UNIT - III**

Decision Tree Learning: - Minimum Description Length Principle. Occam's razor. Learning with active queries

**UNIT - IV**

Neural Network Learning: Perceptions and gradient descent back propagation.

**UNIT - V**

Sample Complexity and Over fitting: Errors in estimating means. Cross Validation and jackknifing VC dimension. Irrelevant features: Multiplicative rules for weight tuning.

**UNIT - VI**

Bayesian Approaches: The basics Expectation Maximization. Hidden Markov Models

**UNIT - VII**

Instance-based Techniques: Lazy vs. eager generalization. K nearest neighbor, case- based reasoning.

**UNIT - VIII**

Genetic Algorithms: Different search methods for induction - Explanation-based Learning: using prior knowledge to reduce sample complexity.

**TEXT BOOKS:**

1. Tom Michel, Machine Learning, McGraw Hill, 1997
2. Trevor Hastie, Robert Tibshirani & Jerome Friedman. The Elements of Statistically Learning, Springer Verlag, 2001

**REFERENCE BOOKS:**

1. Machine Learning Methods in the Environmental Sciences, Neural Networks, William W Hsieh, Cambridge Univ Press.
2. Richard o. Duda, Peter E. Hart and David G. Stork, pattern classification, John Wiley & Sons Inc., 2001
3. Chris Bishop, Neural Networks for Pattern Recognition, Oxford University Press, 1995



**JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY  
HYDERABAD**

<b>IV Year B.Tech. CSE - I Sem</b>	<b>L</b>	<b>T/P/D</b>	<b>C</b>
	3	1/-/-	3

**(57056) SOFT COMPUTING  
(ELECTIVE – II)**

**UNIT-I**

AI Problems and Search: AI problems, Techniques, Problem Spaces and Search, Heuristic Search Techniques- Generate and Test, Hill Climbing, Best First Search Problem reduction.

**UNIT-II**

Constraint Satisfaction and Means End Analysis. Approaches to Knowledge Representation- Using Predicate Logic and Rules.

**UNIT-III**

Artificial Neural Networks: Introduction, Basic models of ANN, important terminologies, Supervised Learning Networks, Perceptron Networks, Adaptive Linear Neuron, Backpropagation Network.

Associative Memory Networks. Training Algorithms for pattern association, BAM and Hopfield Networks.

**UNIT-IV**

Unsupervised Learning Network- Introduction, Fixed Weight Competitive Nets, Maxnet, Hamming Network, Kohonen Self-Organizing Feature Maps, Learning Vector Quantization., Counter Propagation Networks.

**UNIT-V**

Adaptive Resonance Theory Networks. Special Networks-Introduction to various networks.

Introduction to Classical Sets ( crisp Sets)and Fuzzy Sets- operations and Fuzzy sets. Classical Relations.

**UNIT-VI**

Fuzzy Relations- Cardinality, Operations, Properties and composition.

Tolerance and equivalence relations.

Membership functions- Features, Fuzzification, membership value assignments, Defuzzification.

**UNIT-VII**

Fuzzy Arithmetic and Fuzzy Measures, Fuzzy Rule Base and Approximate Reasoning Fuzzy Decision making.

**UNIT-VIII**

Fuzzy Logic Control Systems. Genetic Algorithm- Introduction and basic operators and terminology.Applications: Optimization of TSP, Internet Search Technique

**TEXT BOOKS:**

1. Principles of Soft Computing- S N Sivanandam, S N Deepa, Wiley India, 2007
2. Soft Computing and Intelligent System Design -Fakhreddine O Karray, Clarence D Silva,. Pearson Edition, 2004.

**REFERENCES:**

1. Computational Intelligence, Amit Konar, Springer.
2. Artificial Intelligence and Soft Computing- Behavioural and Cognitive Modelling of the Human Brain- Amit Konar, CRC press, Taylor and Francis Group.
3. Artificial Intelligence – Elaine Rich and Kevin Knight, TMH, 1991, 2nd Edition.
4. Artificial Intelligence – Patric Henry Winston – Third Edition, Pearson Education.
5. A first course in Fuzzy Logic- Hung T Nguyen and Elbert A Walker, CRC. Press Taylor and Francis Group.

**JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY  
HYDERABAD**

<b>IV Year B.Tech. CSE - I Sem</b>	<b>L</b>	<b>T/P/D</b>	<b>C</b>
	3	1/-/-	3

**(57057) INFORMATION RETRIEVAL SYSTEMS  
(ELECTIVE – II)**

**UNIT I**

Introduction, Definition, Objectives, Functional Overview, Relationship to DBMS, Digital libraries and Data Warehouses, Information Retrieval System Capabilities - Search, Browse, Miscellaneous.

**UNIT II**

Cataloging and Indexing: Objectives, Indexing Process, Automatic Indexing, Information Extraction, Data Structures: Introduction, Stemming Algorithms, Inverted file structures, N-gram data structure, PAT data structure, Signature file structure, Hypertext data structure.

**UNIT III**

Automatic Indexing: Classes of automatic indexing, Statistical indexing, Natural language, Concept indexing, Hypertext linkages

**UNIT IV**

Document and Term Clustering: Introduction, Thesaurus generation, Item clustering, Hierarchy of clusters.

**UNIT V**

User Search Techniques: Search statements and binding, Similarity measures and ranking, Relevance feedback, Selective dissemination of information search, Weighted searches of Boolean systems, Searching the Internet and hypertext.

Information Visualization: Introduction, Cognition and perception, Information visualization technologies.

**UNIT VI**

Text Search Algorithms: Introduction, Software text search algorithms,

Hardware text search systems.

Information System Evaluation: Introduction, Measures used in system evaluation, Measurement example – TREC results.

**UNIT VII**

Multimedia Information Retrieval – Models and Languages – Data Modeling, Query Languages, Indexing and Searching.

**UNIT VIII**

Libraries and Bibliographical Systems – Online IR Systems, OPACs, Digital Libraries.

**TEXTBOOKS**

1. Information Storage and Retrieval Systems: Theory and Implementation By Kowalski, Gerald, Mark T Maybury, Springer.
2. Modern Information Retrieval By Ricardo Baeza-Yates, Pearson Education, 2007.
3. Information Retrieval: Algorithms and Heuristics By David A Grossman and Ophir Frieder, 2<sup>nd</sup> Edition, Springer.

**REFERENCE BOOKS**

1. Information Retrieval Data Structures and Algorithms By William B Frakes, Ricardo Baeza-Yates, Pearson Education, 1992.
2. Information Storage & Retrieval By Robert Korfhage – John Wiley & Sons.
3. Introduction to Information Retrieval By Christopher D. Manning and Prabhakar Raghavan, Cambridge University Press, 2008.
4. Natural Language Processing and Information Retrieval. T.Siddiqui and U.S.Tiwary, Oxford Univ. Press.



## JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY

## HYDERABAD

IV Year B.Tech. CSE - I Sem

L	T/P/D	C
0	-/3/-	2

## (57609) LINUX PROGRAMMING AND DATA MINING LAB

- LINUX PROGRAMMING:
  - Note: Use Bash for Shell scripts.
1. Write a shell script that accepts a file name, starting and ending line numbers as arguments and displays all the lines between the given line numbers.
  2. Write a shell script that deletes all lines containing a specified word in one or more files supplied as arguments to it.
  3. Write a shell script that displays a list of all the files in the current directory to which the user has read, write and execute permissions.
  4. Write a shell script that receives any number of file names as arguments checks if every argument supplied is a file or a directory and reports accordingly. Whenever the argument is a file, the number of lines on it is also reported.
  5. Write a shell script that accepts a list of file names as its arguments, counts and reports the occurrence of each word that is present in the first argument file on other argument files.
  6. Write a shell script to list all of the directory files in a directory.
  7. Write a shell script to find factorial of a given integer.
  8. Write an awk script to count the number of lines in a file that do not contain vowels.
  9. Write an awk script to find the number of characters, words and lines in a file.
  10. Write a c program that makes a copy of a file using standard I/O and system calls.
  11. Implement in C the following Unix commands using System calls  
A. cat            B. ls            C. mv
  12. Write a program that takes one or more file/directory names as command line input and reports the following information on the file.

- A. File type.
  - B. Number of links.
  - C. Time of last access.
  - D. Read, Write and Execute permissions.
13. Write a C program to emulate the Unix ls -l command.
  14. Write a C program to list for every file in a directory, its inode number and file name.
  15. Write a C program that demonstrates redirection of standard output to a file. Ex: ls > fl.
  16. Write a C program to create a child process and allow the parent to display "parent" and the child to display "child" on the screen.
  17. Write a C program to create a Zombie process.
  18. Write a C program that illustrates how an orphan is created.
  19. Write a C program that illustrates how to execute two commands concurrently with a command pipe. Ex:- ls -l | sort
  20. Write C programs that illustrate communication between two unrelated processes using named pipe.
  21. Write a C program to create a message queue with read and write permissions to write 3 messages to it with different priority numbers.
  22. Write a C program that receives the messages (from the above message queue as specified in (21)) and displays them.
  23. Write a C program to allow cooperating processes to lock a resource for exclusive use, using a) Semaphores, b) flock or lockf system calls.
  24. Write a C program that illustrates suspending and resuming processes using signals.
  25. Write a C program that implements a producer-consumer system with two processes. (using Semaphores).
  26. Write client and server programs (using c) for interaction between server and client processes using Unix Domain sockets.
  27. Write client and server programs (using c) for interaction between server and client processes using Internet Domain sockets.
  28. Write a C program that illustrates two processes communicating using shared memory.

## TEXT BOOKS:

1. Advanced Unix Programming, N.B.Venkateswarulu, BS Publications.



2. Unix and Shell programming, B.A.Forouzan and R.F.Gilberg, Cengage Learning.
3. Unix and Shell Programming, M.G. Venkatesh Murthy, Pearson Education, 2005.
4. Unix Shells by Example, 4th Edition, Ellie Quigley, Pearson Education.
5. Sed and Awk, O.Dougherty&A.Robbins, 2<sup>nd</sup> edition, SPD.

### Data Mining Lab:

#### Credit Risk Assessment

Description: The business of banks is making loans. Assessing the credit worthiness of an applicant is of crucial importance. You have to develop a system to help a loan officer decide whether the credit of a customer is good, or bad. A bank's business rules regarding loans must consider two opposing factors. On the one hand, a bank wants to make as many loans as possible. Interest on these loans is the banks profit source. On the other hand, a bank cannot afford to make too many bad loans. Too many bad loans could lead to the collapse of the bank. The bank's loan policy must involve a compromise: not too strict, and not too lenient.

To do the assignment, you first and foremost need some knowledge about the world of credit. You can acquire such knowledge in a number of ways.

1. Knowledge Engineering. Find a loan officer who is willing to talk. Interview her and try to represent her knowledge in the form of production rules.
2. Books. Find some training manuals for loan officers or perhaps a suitable textbook on finance. Translate this knowledge from text form to production rule form.
3. Common sense. Imagine yourself as a loan officer and make up reasonable rules which can be used to judge the credit worthiness of a loan applicant.
4. Case histories. Find records of actual cases where competent loan officers correctly judged when, and when not to, approve a loan application.

### The German Credit Data:

Actual historical credit data is not always easy to come by because of confidentiality rules. Here is one such dataset, consisting of 1000 actual cases collected in Germany. credit dataset (original) Excel spreadsheet version of the German credit data. (Down load from web)

In spite of the fact that the data is German, you should probably make use of it for this assignment. (Unless you really can consult a real loan officer !)

A few notes on the German dataset

- DM stands for Deutsche Mark, the unit of currency, worth about 90 cents Canadian (but looks and acts like a quarter).
- owns\_telephone. German phone rates are much higher than in Canada so fewer people own telephones.
- foreign\_worker. There are millions of these in Germany (many from Turkey). It is very hard to get German citizenship if you were not born of German parents.
- There are 20 attributes used in judging a loan applicant. The goal is the classify the applicant into one of two categories, good or bad.

Subtasks : (Turn in your answers to the following tasks)

1. List all the categorical (or nominal) attributes and the real-valued attributes separately.
2. What attributes do you think might be crucial in making the credit assesment ? Come up with some simple rules in plain English using your selected attributes.
3. One type of model that you can create is a Decision Tree - train a Decision Tree using the complete dataset as the training data. Report the model obtained after training.
4. Suppose you use your above model trained on the complete dataset, and classify credit good/bad for each of the examples in the dataset. What % of examples can you classify correctly ? (This is also called testing on the training set) Why do you think you cannot get 100 % training accuracy ?
5. Is testing on the training set as you did above a good idea ? Why orWhy not ?



6. One approach for solving the problem encountered in the previous question is using cross-validation ? Describe what is cross-validation briefly. Train a Decision Tree again using cross-validation and report your results. Does your accuracy increase/decrease ? Why ? (10 marks)
7. Check to see if the data shows a bias against "foreign workers" (attribute 20), or "personal-status" (attribute 9). One way to do this (perhaps rather simple minded) is to remove these attributes from the dataset and see if the decision tree created in those cases is significantly different from the full dataset case which you have already done. To remove an attribute you can use the preprocess tab in Weka's GUI Explorer. Did removing these attributes have any significant effect? Discuss.
8. Another question might be, do you really need to input so many attributes to get good results? Maybe only a few would do. For example, you could try just having attributes 2, 3, 5, 7, 10, 17 (and 21, the class attribute (naturally)). Try out some combinations. (You had removed two attributes in problem 7. Remember to reload the arff data file to get all the attributes initially before you start selecting the ones you want.)
9. Sometimes, the cost of rejecting an applicant who actually has a good credit (case 1) might be higher than accepting an applicant who has bad credit (case 2). Instead of counting the misclassifications equally in both cases, give a higher cost to the first case (say cost 5) and lower cost to the second case. You can do this by using a cost matrix in Weka. Train your Decision Tree again and report the Decision Tree and cross-validation results. Are they significantly different from results obtained in problem 6 (using equal cost)?
10. Do you think it is a good idea to prefer simple decision trees instead of having long complex decision trees ? How does the complexity of a Decision Tree relate to the bias of the model ?
11. You can make your Decision Trees simpler by pruning the nodes. One approach is to use Reduced Error Pruning - Explain this idea briefly. Try reduced error pruning for training your Decision Trees

- using cross-validation (you can do this in Weka) and report the Decision Tree you obtain ? Also, report your accuracy using the pruned model. Does your accuracy increase ?
12. (Extra Credit): How can you convert a Decision Trees into "if-then-else rules". Make up your own small Decision Tree consisting of 2-3 levels and convert it into a set of rules. There also exist different classifiers that output the model in the form of rules - one such classifier in Weka is rules.PART, train this model and report the set of rules obtained. Sometimes just one attribute can be good enough in making the decision, yes, just one ! Can you predict what attribute that might be in this dataset ? OneR classifier uses a single attribute to make decisions (it chooses the attribute based on minimum error). Report the rule obtained by training a one R classifier. Rank the performance of j48, PART and oneR.

**Task Resources:**

- Andrew Moore's Data Mining Tutorials (See tutorials on Decision Trees and Cross Validation)
- Decision Trees (Source: Tan, MSU)
  - Tom Mitchell's book slides (See slides on Concept Learning and Decision Trees)
  - Weka resources:
    - Introduction to Weka (html version) (download ppt version)
    - Download Weka
    - Weka Tutorial
    - ARFF format
    - Using Weka from command line



## JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY

HYDERABAD

IV Year B.Tech. CSE - I Sem

L	T/P/D	C
0	-/3/-	2

## (57610) CASE TOOLS AND SOFTWARE TESTING LAB

## Case Tools Lab

Students are divided into batches of 5 each and each batch has to draw the following diagrams using UML for an ATM system whose description is given below.

UML diagrams to be developed are:

- |                      |                           |
|----------------------|---------------------------|
| 1. Use Case Diagram. | 2. Class Diagram.         |
| 3. Sequence Diagram. | 4. Collaboration Diagram. |
| 5. State Diagram     | 6. Activity Diagram.      |
| 7. Component Diagram | 8. Deployment Diagram.    |
| 9. Test Design.      |                           |

**Description for an ATM System**

The software to be designed will control a simulated automated teller machine (ATM) having a magnetic stripe reader for reading an ATM card, a customer console (keyboard and display) for interaction with the customer, a slot for depositing envelopes, a dispenser for cash (in multiples of Rs. 100, Rs. 500 and Rs. 1000), a printer for printing customer receipts, and a key-operated switch to allow an operator to start or stop the machine. The ATM will communicate with the bank's computer over an appropriate communication link. (The software on the latter is not part of the requirements for this problem.)

The ATM will service one customer at a time. A customer will be required to insert an ATM card and enter a personal identification number (PIN) - both of which will be sent to the bank for validation as part of each transaction. The customer will then be able to perform one or more transactions. The card will be retained in the machine until the customer indicates that he/she desires no further transactions, at which point it will be returned - except as noted below.

The ATM must be able to provide the following services to the customer:

1. A customer must be able to make a cash withdrawal from any suitable account linked to the card, in multiples of Rs. 100 or Rs. 500 or Rs. 1000. Approval must be obtained from the bank before cash is dispensed.
2. A customer must be able to make a deposit to any account linked to the card, consisting of cash and/or checks in an envelope. The customer will enter the amount of the deposit into the ATM, subject to manual verification when the envelope is removed from the machine by an operator. Approval must be obtained from the bank before physically accepting the envelope.
3. A customer must be able to make a transfer of money between any two accounts linked to the card.
4. A customer must be able to make a balance inquiry of any account linked to the card.
5. A customer must be able to abort a transaction in progress by pressing the Cancel key instead of responding to a request from the machine.

The ATM will communicate each transaction to the bank and obtain verification that it was allowed by the bank. Ordinarily, a transaction will be considered complete by the bank once it has been approved. In the case of a deposit, a second message will be sent to the bank indicating that the customer has deposited the envelope. (If the customer fails to deposit the envelope within the timeout period, or presses cancel instead, no second message will be sent to the bank and the deposit will not be credited to the customer.)

If the bank determines that the customer's PIN is invalid, the customer will be required to re-enter the PIN before a transaction can proceed. If the customer is unable to successfully enter the PIN after three tries, the card will be permanently retained by the machine, and the customer will have to contact the bank to get it back.

If a transaction fails for any reason other than an invalid PIN, the ATM will display an explanation of the problem, and will then ask the customer whether he/she wants to do another transaction.

The ATM will provide the customer with a printed receipt for each successful



transaction

The ATM will have a key-operated switch that will allow an operator to start and stop the servicing of customers. After turning the switch to the “on” position, the operator will be required to verify and enter the total cash on hand. The machine can only be turned off when it is not servicing a customer. When the switch is moved to the “off” position, the machine will shut down, so that the operator may remove deposit envelopes and reload the machine with cash, blank receipts, etc.

### Software Testing Lab

#### List of Experiments

1. Write programs in ‘C’ Language to demonstrate the working of the following constructs:  
i) do...while ii) while...do iii) if...else iv) switch v) for
2. “A program written in ‘C’ language for Matrix Multiplication fails”  
Introspect the causes for its failure and write down the possible reasons for its failure.
3. Take any system (e.g. ATM system) and study its system specifications and report the various bugs.
4. Write the test cases for any known application (e.g. Banking application)
5. Create a test plan document for any application (e.g. Library Management System)
6. Study of any testing tool (e.g. Win runner)
7. Study of any web testing tool (e.g. Selenium)
8. Study of any bug tracking tool (e.g. Bugzilla, bugbit)
9. Study of any test management tool (e.g. Test Director)
10. Study of any open source-testing tool (e.g. Test Link)
11. Take a mini project (e.g. University admission, Placement Portal) and execute it. During the Life cycle of the mini project create the various testing documents\* and final test report document.

**\*Note:** To create the various testing related documents refer to the text “Effective Software Testing Methodologies by William E. Perry”