Bachelor of Technology (Computer Science & Engineering)
Credit Based Scheme of Studies/Examination
Semester VII

S. No.	Course No.	Subject	L:T:P	Hours/ Week	Credits	Examination Schedule				Duration of Exam (Hrs.)
						Major Test	Minor Test	Practical	Total	
1	CSE 401	Unix & Linux Programming	4:0:0	4	4.0	75	25	0	100	3
2	CSE 403	Computer Graphics and Animation	4:0:0	4	4.0	75	25	0	100	3
3	PE-I	Elective* – I	3:0:0	3	3.0	75	25	0	100	3
4	PE-II	Elective* – II	3:0:0	3	3.0	75	25	0	100	3
5	CSE 405	Computer Graphics Lab	0:0:2	2	1.0	0	40	60	100	3
6	CSE 407	Project-1**	0:0:8	8	4.0	0	100	100	200	3
7	CSE 409	Unix & Linux Programming Lab	0:0:2	2	1.0	0	40	60	100	3
8	CSE 411	Seminar	0:0:2	2	1.0	0	100	0	100	
9	CSE 413	Industrial Training (Viva-Voce)***					100	0	100	
		Total		**28**	**21.0**	**300**	**480**	**220**	**1000**	

Code	PE-I	Code	PE-II
CSE-415	Object Oriented Software Engineering	CSE-421	Agile Software Engineering
CSE-417	Cyber Security	CSE-423	Big Data and Analytics
CSE-419	Cryptography & Information Security	CSE-425	Expert Systems

Note:
***The students will choose any two departmental electives courses out of the given elective list in 7th Semester.**
****Project should be initiated in 7th semester beginning, and should be completed by the end of 8th semester with good Report and power-point Presentation etc.**
*****4-6 weeks hand on training to be done after 7th Semester Exams.**

CSE-401	Unix & Linux Programming						
Lecture	Tutorial	Practical	Credit	Major Test	Minor Test	Total	Time
4	0	0	4	75	25	100	3 Hrs.
Purpose	Introduces commands and numerous programming concepts and application domains to cover important topics for implementation of the Unix programming concepts.						
Course Outcomes (CO)							
CO1	To learn basic and advanced Unix Commands.						
CO2	Expose the role of filters and file compression techniques.						
CO3	To explore knowledge of programming language development tools.						
CO4	To expand knowledge of Unix/Linux system administration and networking.						

Unit I: Basic Command Usage

Linux Startup: User accounts, accessing Linux - starting and shutting processes, Logging in and Logging out, Unix commands like zip, unzip, pack, unpack, compress, uncompress, Shell Programming, Unix file system: Linux/Unix files, i-nodes and structure, file system related commands, Shell as command processor, shell variables, creating command substitution, scripts, functions, conditionals, loops, customizing environment

Unit II: Filters and File Compression

Regular Expressions and Filters: Introducing regular expressions patterns, syntax, character classes, quantifiers, introduction to grep, egrep, sed, programming with awk and perl, File Compression Techniques: data redundancy elimination using fingerprint generation deduplication and data similarities removal using delta techniques for data reduction storage, parallel compression with Xdelta utility.

Unit III: Program Development Tools

The C Environment: C compiler, vi editor, compiler options, managing projects, memory management, use of makefile, cmake, dependency calculations, memory management – static and dynamic memory, static and dynamic libraries, dynamic loader, debugging tools like gdb, fixed-size and variable-size blocks of data files chunks divisor chunking techniques like Frequency Based Chunking and Content Defined Chunking Unix based open source coding.

Unit IV: Process Control

Processes in Linux: Processes, starting and stopping processes, initialization processes, rc and init files, job control - at, batch, cron, time, network files, security, privileges, authentication, password administration, archiving, Signals and signal handlers, Threading, Linux I/O system, Networking tools like ping, telnet, ftp, route, Firewalls, Backup and Restore tar, cpio, dd.
Case Study: PCOMPRESS open source free software

Text Books:

1. John Goerzen: Linux Programming Bible, IDG Books, New Delhi, 2014.
2. Sumitabha Das: Unix – Concept and Applications, Fourth Edition TMH, 2015.
3. Neil Matthew, Richard Stones: Beginning Linux Programming, 4th. Edition, Wrox-Shroff, 2011.
4. Welsh & Kaufmann: Running Linux, O'Reiley & Associates, 2013.

Reference Book:

1. B.M. Harwani, *Unix and Shell Programming*, Oxford University Press, 2013.

CSE-403	**Computer Graphics and Animation**						
Lecture	**Tutorial**	**Practical**	**Credit**	**Major Test**	**Minor Test**	**Total**	**Time**
4	**0**	**0**	**4**	**75**	**25**	**100**	**03 Hrs.**
Purpose	**Introduces Computer Graphics that help in designing different kinds of static and movable objects.**						
Course Outcomes (CO)							
CO1	Explore the background and standard line and circle drawing algorithms.						
CO2	Exposure of various transformation approaches and its comparative analysis.						
CO3	Illustrate Projection and clipping with explore different techniques.						
CO4	Apply design principles to create different curves and explore hidden lines and surface techniques.						

Unit 1

Computer Graphics applications, Display Devices, Point & Positioning Devices, Plotting Techniques for point and Line, Line drawing algorithms: DDA, Bresenhams's Circle drawing algorithms, Filled area algorithms: Scan line, Polygon filling algorithms, Boundary filled algorithms.

Unit-2

Window to view port transformation, Window to view port mapping, Two Dimensional transformation: translation, scaling, rotation, reflection and Shear, Homogeneous Coordinate system.

3-D transformation: Rotation, Shear, translation, Numerical Problems of transformation viewing pipeline.

Unit-3

Clipping: Point & Line clipping algorithm, 4-bit code algorithm, Cohen-Sutherland Line clipping algorithms, Liang-Barsky line clipping algorithms. Polygon clipping: Sutherland-Hodgeman Polygon clipping algorithm. Curve clipping, Text clipping.

Projection: Parallel, Perspective, Vanishing Points.

Unit-4

Representation of 3-D Curves and Surfaces: interpolation and approximation alpines, parametric conditions, Geometric continuity conditions, Beizer curves and surfaces: properties of beizer curves, beizer surfaces.

Hidden Surfaces removal: Hidden surface elimination, depth buffer algorithm, scan line coherence and area coherence algorithm, priority algorithm

Text Books

1. Donald Hearn & M.Pauline Baker, Computer Graphics, 2nd Edition, Pearson Education.
2. William M. Newmann & Robert F. Sproull, Principles of Interactive Computer Graphics, Tata McGraw-Hill Second Edition, New Delhi, India.
3. Zhigang Xiang & Roy A Plastock , Computer Graphics, Second Edition, Schaum's Outline, Tata McGraw Hill Education Private Limited, New Delhi, India.

References Book

1. Foley, van Dam, Feiner, and Hughes. Computer Graphics: Principles and Practice, 3rd edition in C.

CSE-415	**Object Oriented Software Engineering**						
Lecture	**Tutorial**	**Practical**	**Credit**	**Major Test**	**Minor Test**	**Total**	**Time**
3	**0**	**0**	**3**	**75**	**25**	**100**	**3 Hrs.**
Purpose	**To provide the thorough knowledge to use the concepts and their design attributes for Object Oriented Software Engineering approaches and platforms to solve real time problems.**						
Course Outcomes (CO)							
CO1	To learn the basic concepts of object oriented systems and software engineering.						
CO2	To get exposure of various object modeling methodologies, tools for analyzing and designing software based systems using UML.						
CO3	To explore problems using Use Cases, analyzing relations, responsibilities and collaborations among classes and their behavior in problem domain.						
CO4	To evaluate object oriented design processes using models, design patterns, interfaces designs and communication mechanisms for performing required tasks.						

Unit - I

An Overview of Object-Oriented system Development, Objects Basis, Class Hierarchy, Inheritance, Polymorphism, Object Relationships and Associations, Aggregations and Object Containment, Object Persistence, Meta-Classes, Object Oriented Systems Development Life Cycle: Software Development Process, Object Oriented Systems Development: A Use-Case Driven Approach.

Unit - II

Object Oriented Methodologies: Rumbaugh Methodology, Jacobson Methodology, Booch Methodology, Patterns, Frameworks, The Unified approach, Unified Modeling Language (UML)

Unit - III

Object Oriented Analysis Process, Use Case Driven Object Oriented Analysis, Use Case Model, Object Analysis: Classification, Classification Theory, Approaches for identifying classes, Responsibilities and Collaborators, Identifying Object Relationships, Attributes and Methods: Associations, Super-Sub Class relationships, A-Part-of-Relationships-Aggregation, Class Responsibilities, Object Responsibilities.

Unit - IV

Object Oriented Design process and Design Axioms, Corollaries, Design Patterns, Designing Classes: Object Oriented Design Philosophy, UML Object Constraint Language, Designing Classes: The Process, Class Visibility, Refining Attributes, Designing Methods and Protocols, Packages and Managing classes, View Layer: Designing Interface objects, Designing View layer Classes, Macro and Micro Level Interface Design Process.

TEXT BOOKS:

1. Ali Bahrami, Object Oriented Systems Development, McGraw Hill Publishing Company Limited, New Delhi, 2013.
2. Rumbaugh *et al.*, Object Oriented Modeling and Design, PHI, 2006.
3. Robert Laganière and Timothy C. Lethbridge, Object-Oriented Software Engineering: Practical Software Development, McGraw-Hill Publishing Company Limited, New Delhi, Sixth Print 2008.

REFERENCE BOOKS:

1. Ivar Jacobson, Magnos Christerson, Patrick Jonsson, Gunnar Overgaard, Object-oriented Software Engineering: A Use Case Driven Approach, Pearson Education, New Delhi, Seventh Edition Reprint, 2009.
2. David C. Kung, Object-Oriented Software Engineering: An Agile Unified Methodology, McGraw-Hill Publishing Company Limited, New Delhi, 2013.
3. Bernd Bruegge, Allen H. Dutoit, Object-Oriented Software Engineering Using UML, Patterns, and Java: Pearson New International, Third Edition, 2013.

CSE-417	**Cyber Security**						
Lecture	**Tutorial**	**Practical**	**Credit**	**Major Test**	**Minor Test**	**Total**	**Time**
3	**-**	**-**	**3**	**75**	**25**	**100**	**3 Hrs.**
Purpose	To gain a broad understanding in order to get predictive ways out related to cyber security.						
Course Outcomes							
CO1	To facilitate the basic knowledge of cyber security.						
CO2	To explore and sort issues related to different types of activities in cyber crime.						
CO3	To get enable to fix the various cyber attacks.						
CO4	To deal with the digital forensics and related scenarios of cyber crimes.						

Unit I

Introduction: Introduction and Overview of Cyber Crime, Nature and Scope of Cyber Crime, Types of Cyber Crime: crime against individual, Crime against property, Cyber extortion, Drug trafficking, cyber terrorism.

Need for Information security, Threats to Information Systems, Information Assurance, Cyber Security, and Security Risk Analysis.

Unit 2

Cyber Crime Issues: Unauthorized Access to Computers, Computer Intrusions, Viruses and Malicious Code, Internet Hacking and Cracking, Virus and worms, Software Piracy, Intellectual Property, Mail Bombs, Exploitation, Stalking and Obscenity in Internet, Password Cracking, Steganography, Key loggers and Spyware, Trojan and backdoors, phishing, DOS and DDOS attack, SQL injection, Buffer Overflow.

Unit 3

Introduction to cyber attacks: passive attacks, active attacks.

Cyber crime prevention methods, Application security (Database, E-mail and Internet), Data Security Considerations-Backups, Archival Storage and Disposal of Data, Security Technology-Firewall and VPNs, Intrusion Detection, Access Control, Hardware protection mechanisms, OS Security

Unit 4

Digital Forensics: Introduction to Digital Forensics, historical background of digital forensics, Forensic Software and Hardware, need for computer forensics science, special tools and techniques digital forensic life cycle, challenges in digital forensic.

Law Perspective: Introduction to the Legal Perspectives of Cybercrimes and Cyber security, Cybercrime and the Legal Landscape around the World, Why Do We Need Cyber laws, The Indian IT Act, Cybercrime Scenario in India, Digital Signatures and the Indian IT Act, Cybercrime and Punishment.

TEXT BOOKS:

1. Nelson Phillips and Enfinger Steuart, “Computer Forensics and Investigations”, Cengage Learning, New Delhi, 2009.

REFERENCE BOOKS:

1. Robert M Slade,” Software Forensics”, Tata McGraw - Hill, New Delhi, 2005.
2. Sunit Belapure and Nina Godbole, “Cyber Security: Understanding Cyber Crimes, Computer Forensics And Legal Perspectives”, Wiley India Pvt. Ltd.

CSE-419	Cryptography and Information Security						
Lecture	Tutorial	Practical	Credit	Major Test	Minor Test	Total	Time
3	0	0	3	75	25	100	3 Hrs.
Purpose	The course will be useful for students who plan to do research/product development/analysis in areas related to secure computing in their career.						
Course Outcomes (CO)							
CO1	To learn basics of network security and cryptography.						
CO2	Exposing the knowledge about network authentication mechanism, with security algorithms.						
CO3	To explore the knowledge of key exchange protocols.						
CO4	To realize the effect on digitized security.						

Unit 1: Basics of Cryptography

Introduction to cryptography, security threats, types of cryptography, Classical cryptography and their cryptanalysis, perfect secrecy, Shannon's theorem, stream ciphers, Security attacks

Unit 2: Authentication Mechanism and Security Algorithms

Access control mechanism, Discretionary v/s mandatory access control, CPA-secure encryption, Pseudorandom permutations, practical block ciphers (3-DES, AES), RSA, modes of operation, MACs, Hash functions-Tiger Hash, Gear hash, pseudorandom generators , Public key infrastructure.

Unit 3: Key Exchange Protocols

CCA-secure encryption, Diffie-Hellman key exchange, Public key crypto systems (El Gamal, Paillier, Rabin, Goldwasser-Micali), Key exchange protocols, example protocol such as PGP, Kerberos, IPSEC/VPN, SSL, S/MIME etc., PKCSv1.5.

Unit 4: Digitized Security

Digital signatures,-MD5, SHA1, Rabin Finger Print, digital certificates, DSS, firewall and intrusion detection systems, Byzantine agreement, secure multiparty computation, interactive proof systems

Text Books:

1. Y. Lindell and J. Katz. Introduction to Modern Cryptography. MIT press, 2012.
2. OedGoldreich. Foundations of Modern cryptography: Parts I and II, Cambridge Press, 2011.
3. A. Menezes, P.C. Van Oorschot and S.A. Vanstone. Handbook of Applied Cryptography, CRC Press, 2010.
4. William Stalling, Cryptography and Network Security: Pearson Education, 2013.

Reference Books:

1. Michael EWhitman & Herbert J. Mattord, Principles of Information Security, Vikash Publishing House PVT. LTD., New Delhi, 2015.
2. Charles P. Pfleeger, Security in Computing, 4th Edition, Prentice Hall, 2011.
3. Jeff Crume, Inside Internet Security Addison Wesley, 2014.

CSE-421	**Agile Software Engineering**						
Lecture	**Tutorial**	**Practical**	**Credit**	**Major Test**	**Minor Test**	**Total**	**Time**
3	**0**	**0**	**3**	**75**	**25**	**100**	**3 Hrs.**
Purpose	**Introduces the business value of adopting Agile approaches and provide complete understanding of the Agile development practices**						
Course Outcomes (CO)							
CO1	Understand the background and driving forces for taking an Agile approach to software development.						
CO2	Understand the business value of adopting Agile approaches.						
CO3	Drive development with unit tests using Test Driven Development.						
CO4	Apply design principles and refactoring to achieve Agility.						

Unit I: Fundamentals of Agile

The Genesis of Agile, Introduction and background, Agile Manifesto and Principles, Overview of Scrum, Extreme Programming, Feature Driven development, Lean Software Development, Agile project management, Design and development practices in Agile projects, Test Driven Development, Continuous Integration, Refactoring, Pair Programming, Simple Design, User Stories, Agile Testing, Agile Tools

Unit II: Agile Scrum Framework

Introduction to Scrum, Project phases, Agile Estimation, Planning game, Product backlog, Sprint backlog, Iteration planning, User story definition, Characteristics and content of user stories, Acceptance tests and Verifying stories, Project velocity, Burn down chart, Sprint planning and retrospective, Daily scrum, Scrum roles – Product Owner, Scrum Master, Scrum Team, Scrum case study, Tools for Agile project management.

Unit III: Agile Testing

The Agile lifecycle and its impact on testing, Test-Driven Development (TDD), xUnit framework and tools for TDD, Testing user stories - acceptance tests and scenarios, Planning and managing testing cycle, Exploratory testing, Risk based testing, Regression tests, Test Automation, Tools to support the Agile tester.

Unit IV: Agile Software Design and Development

Agile design practices, Role of design Principles including Single Responsibility Principle, Open Closed Principle, Liskov Substitution Principle, Interface Segregation Principles, Dependency Inversion Principle in Agile Design, Need and significance of Refactoring, Refactoring Techniques, Continuous Integration, Automated build tools, Version control.

Text Books:

1. Ken Schawber, Mike Beedle, *Agile Software Development with Scrum*, Pearson publications.
2. Robert C. Martin, *Agile Software Development, Principles, Patterns and Practices*, Prentice Hall.
3. Lisa Crispin, Janet Gregory, *Agile Testing: A Practical Guide for Testers and Agile Teams*, Addison Wesley.

Reference Books:

1. Alistair Cockburn, *Agile Software Development: The Cooperative Game*, Addison Wesley.
2. Mike Cohn, *User Stories Applied: For Agile Software,* Addison Wesley.

CSE-423	**Big Data and Analytics**						
Lecture	**Tutorial**	**Practical**	**Credit**	**Major Test**	**Minor Test**	**Total**	**Time**
3	**0**	**0**	**3**	**75**	**25**	**100**	**3 Hrs.**
Purpose	**To provide knowledge of Big Data Analytics and Distributed File Systems.**						
Course Outcomes (CO)							
CO1	To learn in details the concepts of big data.						
CO2	Expose the criteria of big data analytics and big data storage.						
CO3	To explore knowledge of big data compression techniques.						
CO4	To explore learning of big data tools and state-of-the-art knowledge with implementation for big data.						

Unit I: Big Data Background

Big data definition and features of big data, big data value, development of big data, challenges of big data, NoSQL databases, technologies related to big data including cloud computing, Internet of Things, data center, Hadoop, relationship between IoT and big data, relationship between hadoop and big data, big data generation and acquisition includes data collection, data transmission, data pre-processing, big data applications.

Unit II: Big Data Analytics and Storage

Big data analysis, big data analytic methods and tools, Pig, Hive, Flume, Mahout, Big data storage, distributed storage system for massive data, storage mechanism for big data GFS, HDFS, HBase, MongoDB, Cassandra, big data storage deduplication techniques, fixed-size and variable-size blocks based deduplication, content defined chunking, frequency based chunking, byte and multi-byte indexing techniques, Cloud storage.

Unit III: Big Data Compression

Big data delta compression, Xdelta implementation, Message Digest (MD5), Secure Hash Algorithm (SHA-1/SHA-256), Gear Hash, Tiger Hash, Rabin and Incremental Secure Fingerprint based deduplication, lossless duplicate and similar data elimination approaches, Parallel deduplication and compression using PCOMPRESS, Scalable Decentralized Deduplication Store (SDDS) using Cassandra.

Unit IV: Big Data Processing

Installation procedure with system requirements for Apache Hadoop, Cassandra, Spark, Pig, Hive, HBase, MongoDB large scale distributed storage systems, Map Reduce programming model working, YARN architecture, Apache Pig and Hive architecture, Single node and Multi-nodes Hadoop Cluster Set up and running a Big Data example, NoSQL implementation.

Text Books:

1. "Big Data" by Viktor Mayer-Schönberger, Kenneth Cukier, ISBN:978-0544002692, Eamon Dolan/Houghton Mifflin Harcourt 2013.
2. “Big Data Now”, by O’Reilly Media Inc., ASIN: B0097E4EBQ, O’Reilly 2012.
3. “Hadoop Operation”, by Eric Sammer, ISBN: 978-1449327057, O’Reilly 2012.
4. “MapReduce Design Patterns: Building Effective Algorithms and Analytics for Hadoop and Other Systems”, by Donald Miner, Adam Shook, ISBN:978-1449327170, O’Reilly 2012.

Reference Books:

1. “Programming Hive”, by Edward Capriolo, ISBN: 978-1449319335,O’Reilly 2012.
2. “HBase: the Definitive Guide”, by Lars George, ISBN: 978-1449396107, O’Reilly 2011.
3. “Mahout in Action”, by Sean Owen, Robin Anil, Ted Dunning, Ellen Friedman, ISBN: 978-1935182689, Manning 2011.
4. “Programming Pig”, by Alan Gates, ISBN: 978-1449302641, O’Reilly 2011.
5. “Cassandra, the Definitive Guide”, by Eben Hewitt ISBN: 978-1449390419 O’Reilly 2011.
6. “MongoDB: The Definitive Guide” by Kristina Chodorow, Michael Dirolf, ISBN: 978-1449381561, O’Reilly, 2010.

CSE-425	Expert Systems						
Lecture	Tutorial	Practical	Credit	Major Test	Minor Test	Total	Time
3	0	0	3	75	25	100	3 Hrs.
Purpose	In this course the student will learn the methodologies used to transfer the knowledge of a human expert into an intelligent program that can be used to solve real-time problems.						
Course Outcomes(CO)							
CO1	Examining the fundamentals and terminologies of expert system.						
CO2	To facilitate students to implement various knowledge representation techniques for acquisition and validate various structures in experts system domain.						
CO3	Signifying AI techniques to solve social, industrial and environmental problems.						
CO4	Application of professional aspects in multi-disciplinary approach to meet global standards towards design, realizing and manufacturing.						

Unit-1

Introduction to AI programming languages, Blind search strategies, Breadth first – Depth first – Heuristic search techniques Hill Climbing – Best first – A Algorithms AO* algorithm – game tress, Min-max algorithms, game playing – Alpha beta pruning.

Knowledge representation issues predicate logic – logic programming Semantic nets- frames and inheritance, constraint propagation; Representing Knowledge using rules, Rules based deduction systems.

Unit-2

Introduction to Expert Systems, Architecture of expert system, Representation and organization of knowledge, Basics characteristics, and types of problems handled by expert systems.

Expert System Tools: Techniques of knowledge representations in expert systems, knowledge
engineering, System-building aids, support facilities, stages in the development of expert systems.

Unit-3

Building an Expert System: Expert system development, Selection of tool, Acquiring Knowledge, Building process.

Unit-4

Problems with Expert Systems: Difficulties, common pitfalls in planning, dealing with domain
expert, difficulties during development.

TEXT BOOKS

1. Elain Rich and Kevin Knight, “Artificial Intelligence”, Tata McGraw-Hill, New Delhi, 2008.
2. Waterman D.A., “A Guide to Expert Systems”, Addison Wesley Longman, 1985.

REFERENCE BOOKS

1. Staurt Russel and other Peter Norvig, “Artificial Intelligence – A Modern Approach”, Prentice Hall, 1995.
2. Patrick Henry Winston, “Artificial Intelligence”, Addison Wesley, 1979.
3. Patterson, Artificial Intelligence & Expert System, Prentice Hall India, 1999.
4. Hayes-Roth, Lenat and Waterman: Building Expert Systems, Addison Wesley, 1983.
5. Weiss S.M. and Kulikowski C.A., “A Practical Guide to Designing Expert Systems”, Rowman & Allanheld, New Jersey, 2011.

CSE-405	Computer Graphics Lab						
Lecture	Tutorial	Practical	Credit	Minor Test	Practical	Total	Time
0	0	2	1	40	60	100	3Hrs.
Purpose	To Design and implement various Line and Circle Drawing Algorithms.						
Course Outcomes (CO)							
CO1	To Implement basic algorithms related to Line & Circle Drawing.						
CO2	Implement various Line & Circle Drawing Algorithms.						
CO3	Hands on experiments on 2-D transformations.						
CO4	Conceptual implementation of Clipping and other drawing algorithms..						

List of Practicals

1. Write a program to implement DDA line drawing algorithm.
2. Write a program to implement Bresenham’s line drawing algorithm.
3. Implement the Bresenham’s circle drawing algorithm.
4. Write a program to draw a decagon whose all vertices are connected with every other vertex using lines.
5. Write a program to move an object using the concepts of 2-D transformations.
6. Write a program to implement the midpoint circle drawing algorithm any Object Oriented Programming Language like Python, C++, Java.
7. Implement the line clipping algorithm using any Object Oriented Programming Language like Python, C++, Java.
8. Implement boundary fill algorithm using any Object Oriented Programming Language like Python, C++, Java.
9. Implement the depth buffer algorithm using any Object oriented language like Python, C++, Java.
10. Perform the Polygon Clipping Algorithm using any Object oriented language like Python, C++, Java.
11. Draw a Rectangle using Bresenham’s and DDA Algorithm using any Object oriented language like Python, C++, Java.

CSE-409	Unix & Linux Programming Lab						
Lecture	**Tutorial**	**Practical**	**Credit**	**Minor Test**	**Practical**	**Total**	**Time**
0	**0**	**2**	**1**	**40**	**60**	**100**	**3 Hrs.**
Purpose	**To provide experimental knowledge of Unix & Linux Programs**						
Course Outcomes (CO)							
CO	Exploring knowledge by implementation of programs using UNIX/LINUX.						

List of Practicals

1. Familiarize with Unix/Linux logging/logout and simple commands.
2. Familiarize with vi editor.
3. Using Bash shell develop simple shell programs.
4. Develop advanced shell programs using grep, fgrep & egrep.
5. Compile and debug various C programs using different options.
   Content defined chunking, frequency based chunking, delta/Xdelta, Rabin Fingerprint Generator, Parallel Compression pcompress.
6. Learning of installation and upgradation of Linux operating system.
7. Install Linux on a PC having some other previously installed operating system. All OSs should be usable.
8. As supervisor create and maintain user accounts, learn package installation, taking backups, creation of scripts for file and user management, creation of startup and shutdown scripts using at, cron etc.

Note: At least 5 to 10 more exercises are to be given by the teacher concerned.

Bachelor of Technology (Computer Science & Engineering)
Credit Based Scheme of Studies/Examination
Semester VIII

S. No.	Course No.	Subject	L:T:P	Hours/ Week	Credits	Examination Schedule				Duration of Exam (Hrs.)
						Major Test	**Minor Test**	**Practical**	**Total**	
1	CSE 402	Neural Networks & Fuzzy Logic	4:0:0	4	4.0	75	25	0	100	3
2	PE-III	Elective*-III	4:0:0	4	4.0	75	25	0	100	3
3	PE-IV	Elective* - IV	4:0:0	4	4.0	75	25	0	100	3
4	CSE 404	Mobile Apps Development	4:0:0	4	4.0	75	25	0	100	3
5	CSE 406	Mobile Apps Development Lab	0:0:2	2	1.0	0	40	60	100	3
6	CSE 408	Computer Hardware & Troubleshooting Lab	0:0:2	2	1.0	0	40	60	100	3
7	CSE 410	Project-II	0:0:16	16	8.0	0	100	100	200	3
		Total		**36**	**26.0**	**300**	**280**	**220**	**800**	
8	CSE 424	General Fitness & Professional Aptitude**					100		100	8

Code	PE-I	Code	PE-II
CSE-412	Software Testing	CSE-418	Parallel Computing
CSE-414	Graph Theory	CSE-420	Cloud Computing
CSE-416	Data Mining	CSE-422	Natural Language Processing

Note:
***The students will choose any two departmental electives courses out of the given elective list in VIII Semester.**
****CSE 424 will be non credit subject and only grade will be awarded to student based on his/her performance in the examination.**

CSE-402	**Neural Networks & Fuzzy Logic**						
Lecture	**Tutorial**	**Practical**	**Credit**	**Major Test**	**Minor Test**	**Total**	**Time**
4	**0**	**0**	**4**	**75**	**25**	**100**	**3 Hrs.**
Purpose	**To provide knowledge of various artificial neural networks, fuzzy logic techniques and Genetic Engineering approach for optimization**						
Course Outcomes (CO)							
CO1	To learn the basics of artificial neural networks concepts.						
CO2	Expose detailed explanation of various neural networks architecture.						
CO3	To explore knowledge of special types of Artificial neural networks.						
CO4	To explore fuzzy logic techniques and genetic algorithms in neural networks.						

Unit I: Fundamentals of Artificial Neural Networks

Introduction: Concepts of neural networks, Characteristics of Neural Networks, Applications of Neural Networks. Fundamentals of Neural Networks: The biological prototype, Neuron concept, Single layer Neural Networks, Multi-Layer Neural Networks, terminology, Notation and representation of Neural Networks, Training of Artificial Neural Networks. Representation of perceptron, perceptron learning and training, Classification, linear Separability

Unit II: Neural Networks

Hopfield nets: Structure, training, and applications, Back Propagation: Concept, Applications and Back Propagation Training Algorithms. Counter Propagation Networks: Kohonan Network, Grossberg Layer & Training, applications of counter propagation, Image classification.

Bi-directional Associative Memories: Structure, retrieving a stored association, encoding associations.

Unit III: Special Neural Networks

ART: ART architecture, ART classification operation, ART implementation and characteristics of ART. Image Compression Using ART, Optical Neural Networks: Vector Matrix Multipliers, Hop field net using Electro optical matrix multipliers, Holographic correlator, Optical Hopfield net using Volume Holograms, Cognitrons and Neocognitrons: structure and training.

Unit IV: Fuzzy Logic

Fuzzy Logic: Introduction to Fuzzy Logic, Classical and Fuzzy Sets: Overview of Classical Sets, Membership Function, Fuzzy rule generation, Operations on Fuzzy Sets: Compliment, Intersections, Unions, Combinations of Operations, Aggregation Operations, Fuzzy Arithmetic: Fuzzy Numbers, Linguistic Variables, Arithmetic Operations on Intervals & Numbers, Lattice of Fuzzy Numbers, Fuzzy Equations, Introduction of Neuro-Fuzzy Systems, Architecture of Neuro Fuzzy Networks, Genetic Algorithms: genetic algorithm implementation in problem solving and working of genetic algorithms evolving neural networks, Differential Evolution optimization for engineering problems.

Text Books:

1. Li Min Fu, “Neural Networks in Computer Intelligence”, McGraw-Hill, Inc. 2012.
2. S N Sivanandam, “Neural Networks using MATLAB 6.0”, TMH, 4th. Reprint 2015.
3. S N Sivanandam, “Principles of Soft Computing”, 2nd. Edition, Wiley, Reprint 2014.

Reference Books:

1. Simon Haykin, “Neural Networks: A Comprehensive Foundations”, Prentice-Hall International, New Jersey, 2013.
2. Freeman J.A. & D.M. Skapura, “Neural Networks: Algorithms, Applications and Programming Techniques”, Addison Wesley, Reading, Mass, 2014.

CSE-412	Software Testing						
Lecture	**Tutorial**	**Practical**	**Credit**	**Major Test**	**Minor Test**	**Total**	**Time**
4	**0**	**0**	**4**	**75**	**25**	**100**	**3 Hrs.**
Purpose	To provide an understanding of concepts and techniques for testing software and assuring its quality.						
Course Outcomes (CO)							
CO1	Expose the criteria and parameters for the generation of test cases.						
CO2	Learn the design of test cases and generating test cases.						
CO3	Be familiar with test management and software testing activities.						
CO4	Be exposed to the significance of software testing in web and Object orient techniques.						

UNIT – I

Introduction: Overview of software evolution, SDLC, Testing Process, Terminologies in Testing: Error, Fault, Failure, Verification, Validation, Difference between Verification and Validation, What is software testing and why it is so hard? Test Cases, Test Oracles, Testing Process, Limitations of Testing.

UNIT - II

Functional Testing: Boundary Value Analysis, Equivalence Class Testing, Decision Table Based Testing, Cause Effect Graphing Technique.

Structural Testing: Path testing, DD-Paths, Cyclomatic Complexity, Graph Metrics, Data Flow Testing, Mutation testing.

UNIT - III

Reducing the number of test cases: Prioritization guidelines, Priority category, Scheme, Risk Analysis, Regression Testing, and Slice based testing

Testing Activities: Unit Testing, Levels of Testing, Integration Testing, System Testing, Debugging, Domain Testing.

UNIT - IV

Object oriented Testing: Definition, Issues, Class Testing, Object Oriented Integration and System Testing.

Testing Web Applications: What is Web testing?, User interface Testing, Usability Testing, Security Testing, Performance Testing, Database testing, Post Deployment Testing.

TEXT BOOKS:

1. Naresh Chauhan “Software Testing Principles and Practices” Oxford Publications, 2012.
2. Louise Tamres, “Software Testing”, Pearson Education Asia, 2002.
3. Robert V. Binder, “Testing Object-Oriented Systems-Models, Patterns and Tools”, Addison Wesley, 1999.
4. William Perry, “Effective Methods for Software Testing”, John Wiley & Sons, New York, 1995.

REFERENCE BOOKS:

1. Cem Kaner, Jack Falk, Nguyen Quoc, “Testing Computer Software”, Second Edition, Van Nostrand Reinhold, New York, 1993.
2. K.K. Aggarwal & Yogesh Singh, “Software Engineering”, 2nd Ed., New Age International Publishers, New Delhi, 2005.
3. Boris Beizer, “Software Testing Techniques”, Second Volume, Second Edition, Van Nostrand Reinhold, New York, 1990.
4. Boris Beizer, “Black-Box Testing – Techniques for Functional Testing of Software and Systems”, John Wiley & Sons Inc., New York, 1995.
5. Gopalaswamy Ramesh, Srinivasan Desikan, Software Testing : Principles and Practices, Pearson India, 2005.

CSE-414	Graph Theory						
Lecture	Tutorial	Practical	Credit	Major Test	Minor Test	Total	Time
4	0	0	4	75	25	100	3Hrs.
Purpose	To familiarize the students with the fundamentals of Graph Theory and Graph algorithms.						
Course Outcomes							
CO1	To get enabled about the various concepts of graph theory.						
CO2	To explore different trees, graphs and algorithms.						
CO3	To deal with the concept of planar graph and its related algorithms.						
CO4	To implement the concept of vectors, colouring, covering and partitioning of a graph.						

UNIT- I

Introduction : Graphs, Isomorphism, Sub graphs, some basic properties, various example of graphs & their sub graphs, walks, path & circuits, connected graphs, disconnected graphs and component, directed graphs, types of directed graphs, Euler graphs, various operation on graphs, Hamiltonian paths and circuits, the traveling sales man problem.

UNIT- II

Trees: Trees and fundamental circuits, distance diameters, radius and pendent vertices, rooted and binary trees, on counting trees, spanning trees, fundamental circuits, finding all spanning trees of a graph and a weighted graph, algorithms of primes, Kruskal and Dijkstra Algorithms.

UNIT- III

Fundamentals of Cut sets: Cut sets Cuts sets and cut vertices, some properties, all cut sets in a graph, fundamental circuits and cut sets, connectivity and separability, network flows.

Planar Graphs: Planer graphs, different representation of a planar graph, combinatorial and geometric dual: Kuratowski graphs, detection of planarity, geometric dual, Discussion on criterion of planarity, thickness and crossings.

UNIT- IV

Vector: Vector space of a graph and vectors, basis vector, cut set vector, circuit vector, circuit and cut set subspaces, Matrix representation of graph – Basic concepts; Incidence matrix, Circuit matrix, Path matrix, Cut-set matrix and Adjacency matrix.

Graph Colouring, covering and partitioning: Colouring, covering and partitioning of a graph, chromatic number, chromatic partitioning, chromatic polynomials, matching, covering, four color problem.

Text Books:

1. Deo, N, Graph theory with applications to Engineering and Computer Science, PHI.
2. Gary Chartrand and Ping Zhang, Introduction to Graph Theory, TMH.

Reference Books:

1. Robin J. Wilson, Introduction to Graph Theory, Pearson Education.
2. Harary, F, Graph Theory, Narosa Publication.
3. Bondy and Murthy: Graph theory and application. Addison Wesley.
4. V. Balakrishnan, Schaum's Outline of Graph Theory, TMH.
5. Geir Agnarsson, Graph Theory: Modeling, Applications and Algorithms, Pearson Education.

CSE-416	**Data Mining**						
Lecture	**Tutorial**	**Practical**	**Credit**	**Major Test**	**Minor Test**	**Total**	**Time**
4	**0**	**0**	**4**	**75**	**25**	**100**	**3 Hrs.**
Purpose	**To provide the knowledge of data mining and its techniques.**						
Course Outcomes (CO)							
CO1	To learn data mining concepts in details.						
CO2	Expose the criteria for data generalization.						
CO3	To explore knowledge of mining associations, correlations and classification.						
CO4	To evaluate various types of data mining.						

Unit I: Data Mining and Data Preprocessing

Introduction : Data Mining, Functionalities, Data Mining Systems classification, Integration with Data Warehouse System, Data summarization, data cleaning, data integration and transformation, data reduction. Data Warehouse: Need for Data Warehousing, Paradigm Shift, Business Problem Definition, Operational and Information Data Stores, Data Warehouse Definition and Characteristics, Data Warehouse Architecture and Implementation, OLAP.

Unit II: Data Generalization

Data Mining Primitives, Query Language and System Architecture, Concept Description, Data generalization, Analysis of attribute relevance, Mining descriptive statistical measures in large databases, Data deduplication methodologies.

Unit III: Mining Associations and Correlations

Mining association rules in large databases: Association rule mining, Mining single dimensional boolean association rules from transactional databases, mining multilevel association rules from transaction databases, Relational databases and data warehouses, correlation analysis, classification and prediction, Data redundancy detection and elimination techniques.

Unit IV: Cluster Analysis and Mining

Introduction to cluster analysis, Mining complex type of data: Multidimensional analysis and descriptive mining of complex data objects, Spatial databases, Multimedia databases, Mining time series and sequence data, Mining text databases, Mining World Wide Web, Data Chunking Techniques.

Text Books

1. J.Han, M.Kamber, Data Mining: Concepts and Techniques, Academic Press, Morgan Kanfman Publishers, 2015.
2. Pieter Adrians, Dolf Zantinge, Data Mining, Addison Wesley 2013.
3. C.S.R. Prabhu, Data Ware housing: Concepts, Techniques, Products and Applications, Prentice Hall of India, 2014.

Reference Books

1. Berry and Lin off, Mastering Data Mining: The Art and Science of Customer Relationship Management, John Wiley and Sons, 2012.
2. Seidman, Data Mining with Microsoft SQL Server, Prentice Hall of India, 2016.

CSE-418	**Parallel Computing**						
Lecture	**Tutorial**	**Practical**	**Credit**	**Major Test**	**Minor Test**	**Total**	**Time**
3	**0**	**0**	**3**	**75**	**25**	**100**	**3 Hrs.**
Purpose	**To enable students to compare various architectural taxonomies and design paradigms of parallel computers and computational models, parallelism approaches, performance metrics and techniques to parallelize and schedule loops and their programming constructs.**						
Course Outcomes (CO)							
CO1	Classify various synchronous and asynchronous paradigms of parallel computing as well as identify some of the taxonomies for architectural classification of parallel computers.						
CO2	Compare various parallel computation models and approaches and describe different performance metrics in parallel computers.						
CO3	Distinguish shared memory and distributed memory multiprocessors and explain various parallel programming models and relative advantages and disadvantages of interconnection networks based on network parameters for reliable connections and achieving efficient speed.						
CO4	Examine various techniques of parallelizing loops and sequential programs and scheduling.						

Unit-I

Introduction: The state of computing, system attributes to performance, Paradigms of parallel computing: Synchronous – Vector/ Array, SIMD, systolic, Asynchronous- MIMD, reduction paradigm.

Hardware Taxonomy: Flynn's classification, Feng's classification, handler's classification.
Software taxonomy: Kung's taxonomy.

Unit-II

Abstract parallel computational models: combinational circuits, sorting network, PRAM models, VLSI complexity model, Interconnections RAMs, Parallelism approaches- data parallelism, control parallelism, Conditions of parallelism: Data, control and resource dependencies, Hardware and software parallelism.

Performance metrics: Laws governing performance measurements, Metrics- speedups, efficiency, utilization, communication overheads, single/ multiple program performances.

Unit-III

Parallel processors: taxonomy and topology: shared memory multi processors, distributed memory multicomputer, static and dynamic interconnections.

Parallel programming: shared memory programming, distributed memory programming, object oriented programming, data parallel programming, functional and data flow programming.

Unit-IV

Scheduling and parallelization: Loop parallelization and pipelining-Loop transformation theory, parallelization and wave fronting, tiling and localization, software pipelining, Scheduling parallel programs, program partitioning and scheduling: Grain size, latency, grain packing and scheduling, loop scheduling, Parallelization of sequential programs.

Text Books

1. Kai Hwang and Naresh Jotwani, Advanced Computer Architecture, Second Edition, McGraw Hill, New Delhi, India, 2012.
2. M.J. Quinn, Parallel Computing: Theory and Practice, Second Edition, McGraw Hill, New Delhi, India, 2008.
3. D.Sima, T.Fountain, P.Kasuk, Advanced Computer Architecture-A Design space Approach, Pearson Education, India, 2009.

Reference Books

1. J. L. Hennessy and D. A. Patterson, Computer Architecture: A Quantitative approach, 5th Edition, Morgan Kaufmann/Elsevier-India.
2. T.G.Lewis, Parallel Programming: A machine Independent approach, IEEE Computer Society Press,Los Alamitos, 1994.
3. T.G.Lewis and H. El-Rewini, Introduction to parallel computing, Prentice Hall, New Jersey, 1998.

CSE-420	**Cloud Computing**						
Lecture	**Tutorial**	**Practical**	**Credit**	**Major Test**	**Minor Test**	**Total**	**Time**
4	**0**	**0**	**4**	**75**	**25**	**100**	**03 Hrs.**
Purpose	**To familiar the concepts of cloud services and storage to deploy various resources and arbitrary software.**						
Course Outcomes (CO)							
CO1	Facilitate the basic usage and applicability of computing paradigm.						
CO2	Explore various cloud service and deployment models to utilize different cloud services.						
CO3	To get enabled for various data, scalability & cloud services in order to get efficient database for cloud storage.						
CO4	To deal with various security threats and their controlling mechanism for accessing safe cloud services.						

Unit-1

Overview of Computing Paradigm: Recent trends in Computing, Grid Computing, Cluster Computing, Distributed Computing, Utility Computing, Cloud Computing, evolution of cloud computing, Business driver for adopting cloud computing.

Cloud Computing (NIST Model), History of Cloud Computing, Cloud service providers, Properties, Characteristics & Disadvantages, Pros and Cons of Cloud Computing, Benefits of Cloud Computing, Cloud computing vs. Cluster computing vs. Grid computing, Role of Open Standards.

Unit-2

Cloud Computing Architecture: Cloud computing stack, Comparison with traditional computing architecture (client/server), Services provided at various levels, How Cloud Computing Works, Role of Networks in Cloud computing, protocols used, Role of Web services, Service Models (XaaS) - Infrastructure as a Service (IaaS), Platform as a Service (PaaS), Software as a Service (SaaS), Deployment Models- Public cloud, Private cloud, Hybrid cloud, Community cloud.

Unit-3

Service Management in Cloud Computing: Service Level Agreements (SLAs), Billing & Accounting, comparing Scaling Hardware: Traditional vs. Cloud, Economics of scaling: Benefitting enormously, Managing Data- Looking at Data, Scalability & Cloud Services, Database & Data Stores in Cloud, Large Scale Data Processing.

Case study: Eucalyptus, Microsoft Azure, Amazon EC2.

Unit-4

Cloud Security: Infrastructure Security, Network level security, Host level security, Application level security, Data security and Storage, Data privacy and security Issues, Jurisdictional issues raised by Data location, Identity & Access Management, Access Control, Trust, Reputation, Risk, Authentication in cloud computing, Client access in cloud, Cloud contracting Model, Commercial and business considerations.

Text Books

1. Barrie Sosinsky, Cloud Computing Bible, Wiley-India, 2010.
2. Rajkumar Buyya, James Broberg, Andrzej M. Goscinski, Cloud Computing: Principles and Paradigms, Wiley, 2011.

Reference Books

1. Nikos Antonopoulos, Lee Gillam, Cloud Computing: Principles, Systems and Applications, Springer, 2012.
2. Ronald L. Krutz, Russell Dean Vines, Cloud Security: A Comprehensive Guide to Secure Cloud Computing, Wiley-India, 2010.

CSE-422	**Natural Language Processing**						
Lecture	**Tutorial**	**Practical**	**Credit**	**Major Test**	**Minor Test**	**Total**	**Time**
4	**0**	**0**					**3 Hrs.**
Purpose	To provide the understanding of the mathematical and linguistic foundations underlying approaches to the various areas in NLP.						
Course Outcomes (CO)							
CO1	Be familiar with syntax and semantics in NLP.						
CO2	To implement various concepts of knowledge representation using Prolog.						
CO3	To classify different parsing techniques and understand semantic networks.						
CO4	To identify/explain various applications of NLP.						

Unit-1

Fundamental components of Natural Language Processing: Lexicography, syntax, semantics, prosody, phonology, pragmatic analysis, world knowledge.
Knowledge Representation schemes: Semantic net, Frames, Conceptual Dependency, Scripts.

Unit-2

Representing knowledge using rules: Logic Programming, Introduction to LISP and Prolog, Rules based deduction systems, General concepts in knowledge acquisition.
Syntax Analysis: Formal Languages and grammars, Chomsky Hierarchy, Left- Associative Grammars, ambiguous grammars, resolution of ambiguities.

Unit-3

Computation Linguistics: Recognition and parsing of natural language structures- ATN and RTN, General Techniques of parsing- CKY, Earley and Tomitas algorithm.
Semantics: Knowledge representation, semantics networks logic and inference pragmatics, graph models and optimization.

Unit-4

Applications of NLP: Intelligent work processor, Machine translation, user interfaces, Man-Machine interfaces, natural language querying, tutoring and authoring systems, speech recognition, commercial use of NLP.

Text Books:

1. Daniel Jurafsky, James H. Martin, “Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics and Speech Recognition”, 2nd edition, Pearson Edu., 2013.
2. James Allen, “Natural Language Understanding”, Pearson Education, Second Edition, 2003.

Reference Books:

1. Ivan Bratko, “Prolog: Programming for Artificial Intelligence”, 3rd Edition, Pearson Education, Fifth Impression 2009.
2. G. Gazder, “Natural Language processing in prolog”, Addison Wesley, 1989.

CSE-404	**Mobile Apps Development**						
Lecture	**Tutorial**	**Practical**	**Credit**	**Major Test**	**Minor Test**	**Total**	**Time**
4	**0**	**0**	**4**	**75**	**25**	**100**	**3 Hrs.**
Purpose	**To introduce the concepts of developing the mobile applications.**						
Course Outcomes (CO)							
CO1	Be exposed to technology and Mobile apps development aspects.						
CO2	Be competent with the characterization and architecture of mobile applications.						
CO3	Appreciation of nuances such as native hardware play, location awareness, graphics, and multimedia.						
CO4	Perform testing, signing, packaging and distribution of mobile apps.						

Unit 1: Introduction to Mobility

Mobility landscape, Mobile platforms, Mobile apps development, Overview of Android platform, Setting up the Mobile App Development environment along with an Emulator.

App User Interface Designing – Mobile UI resources (Layout, UI elements, Drawable, Menu).

Unit II: Building blocks of Mobile Apps

Activity- States and Life Cycle, Interaction amongst Activities.

App functionality beyond user interface - Threads, Async task, Services – States and Life Cycle, Notifications, Broadcast receivers, Content provider.

Unit III: Sprucing up Mobile Apps

Graphics and animation – Custom views, Canvas, Animation APIs, Multimedia – Audio/Video playback and record, Location awareness.

Native data handling –file I/O, Shared preferences, Mobile databases such as SQLite, and Enterprise data access (via Internet/Intranet).

Unit IV: Testing Mobile Apps

Debugging mobile apps, White box testing, Black box testing, and test automation of Mobile apps, JUnit for Android.

Text Books:

1. Barry Burd, *Android Application Development All in one for Dummies*, Wiley publications, 2nd Edition 2015.
2. Android Developer Fundamentals Course– Concepts (Learn to develop Android applications) Concepts Reference *Developed by Google Developer Training Team, 2016.*
3. Valentino Lee, Heather Schneider, and Robbie Schell, Mobile Applications: Architecture, Design, and Development, Prentice Hall, 2004.
4. Rick Boyer, Kyle Mew, Android Application Development Cookbook - Second Edition, 2016.

Reference Books:

1. Carmen Delessio, Lauren Darcey, Teach Yourself Android Application Development In 24 Hours , SAMS, 2013.
2. Brian Fling, Mobile Design and Development, O'Reilly Media, 2009.
3. Maximiliano Firtman, Programming the Mobile Web, O'Reilly Media, 2010.
4. Christian Crumlish and Erin Malone, Designing Social Interfaces, O'Reilly Media, 2009.
5. Jerome F. DiMarzio, Beginning Android Programming with Android Studio, 4th edition, 2016.
6. Max Lemann , Android Studio: App Development on Android 6, 2016.

CSE-406	Mobile Apps Development Lab						
Lecture	**Tutorial**	**Practical**	**Credit**	**Minor Test**	**Practical**	**Total**	**Time**
-	**-**	**2**	**1**	**40**	**60**	**100**	**3 Hrs.**
Purpose	Design and Implement various mobile applications using emulators and learn how to Deploy applications to hand-held devices.						
Course Outcomes (CO)							
CO1	Know the components and structure of mobile application development frameworks for Android based mobiles.						
CO2	Understand how to work with various mobile application development frameworks.						
CO3	Learn the basic and important design concepts and issues of development of mobile applications.						
CO4	Understand the capabilities of mobile devices.						

LIST OF PRACTICALS:

1. Develop an application that uses GUI components, Font and Colours
2. Develop an application that uses Layout Managers and event listeners.
3. Develop a native calculator application.
4. Write an application that draws basic graphical primitives on the screen.
5. Implement an application that implements Multi threading
6. Develop a native application that uses GPS location information.
7. Implement an application that writes data to the SD card.
8. Implement an application that creates an alert upon receiving a message.
9. Write a mobile application that creates alarm clock.
10. Develop a sign-in page with appropriate validation.
11. Develop a real life application that makes use of database.

Note: At least 5 to 10 more exercises are to be given by the teacher concerned.

CSE-408	**Computer Hardware & Troubleshooting Lab**						
Lecture	**Tutorial**	**Practical**	**Credit**	**Minor Test**	**Practical**	**Total**	**Time**
0	**0**	**2**	**1**	**40**	**60**	**100**	**3 Hrs.**
Purpose	**To study the current personal computer hardware including personal computer assembly upgrading, setup configuration and troubleshooting.**						
Course Outcomes (CO)							
CO1	To understand the fundamental hardware components that makes up a computer's hardware and the role of each of these components.						
CO2	Assemble/setup and upgrade personal computer hardware.						
CO3	Perform installation, configuration, and upgrading of microcomputer hardware and software.						
CO4	Diagnose and troubleshoot microcomputer systems hardware and software, and other peripheral equipment.						

List of Practicals:

1. To make the comparative study of various motherboards.
2. To study various cables used in computer communication.
3. To study various connections and ports used in computer communication.
4. To study various cards used in a computer System like Ethernet, sound, video card etc.
5. To study different microprocessor like P-IV, dual core, i3, i5, i7 etc.
6. To study SMPS and UPS.
7. To study rotational and loading mechanisms of the following drives:(Floppy disk drive, Hard disk, CD ROM,CD-R/RW,DVD-ROM, DVD recordable drives, DUAL LAYER DVD-R/W)
8. To study monitor and its circuitry (CRT (Cathode Ray Tube), LCD (Liquid Crystal Display), LED (Light-Emitting Diodes), Plasma (OLED).
9. To study different types of printers and its installation.
10. To study working of keyboard and mouse.
11. To assemble a PC and trouble shooting.
12. To install different Operating System and install different hardware components.

Text Books:

1. How Computers WorkBy, Ron White and Timothy Edward Downs, 10th Revised edition, Pearson Education, 2014.
2. Upgrading and Repairing PCs, Scott Mueller, 22nd Edition, Que Publishing, 2015.
3. Learning PC Hardware, Ramesh Bangia, Khanna Book Publishing, 2nd revised edition, 2012.

Reference Book:

1. Pc Hardware: The Complete Reference 1st Edition, Craig Zacker, McGraw Hill Education; 1st edition, 2001.
2. Modern Computer Hardware Course, Manahar Lotia, Pradeep Nair, Payal Lotia, BPB Publications, 2nd Revised Edition, 2007.