

Annexure-3

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

| S. No. | Course No. | Subject | L:T:P | Hours/Week | Examination Schedule | | | | Duration of Exam (Hrs.) |
|--------|------------|------------------------------------|-------|------------|----------------------|------------|------------|-------------|-------------------------|
| | | | | | Major Test | Minor Test | Practical | Total | |
| 1 | CSE 401N | Unix & Linux Programming | 4:0:0 | 4 | 75 | 25 | 0 | 100 | 3 |
| 2 | CSE 403N | Computer Graphics and Animation | 4:0:0 | 4 | 75 | 25 | 0 | 100 | 3 |
| 3 | PE-I | Elective* – I | 3:0:0 | 3 | 75 | 25 | 0 | 100 | 3 |
| 4 | PE-II | Elective* – II | 3:0:0 | 3 | 75 | 25 | 0 | 100 | 3 |
| 5 | CSE 405N | Computer Graphics Lab | 0:0:2 | 2 | 0 | 40 | 60 | 100 | 3 |
| 6 | CSE 407N | Project-I** | 0:0:9 | 9 | 0 | 100 | 100 | 200 | 3 |
| 7 | CSE 409N | Unix & Linux Programming Lab | 0:0:2 | 2 | 0 | 40 | 60 | 100 | 3 |
| 8 | CSE 411N | Seminar | 0:0:2 | 2 | 0 | 100 | 0 | 100 | |
| 9 | CSE 413N | Industrial Training (Viva-Voce)*** | | | | 100 | 0 | 100 | |
| | | Total | | 29 | 300 | 480 | 220 | 1000 | |

| Code | PE-I | Code | PE-II |
|----------|--------------------------------------|----------|----------------------------|
| CSE-415N | Object Oriented Software Engineering | CSE-421N | Agile Software Engineering |
| CSE-417N | Cyber Security | CSE-423N | Big Data and Analytics |

| | | | |
|----------|-------------------------------------|----------|----------------|
| CSE-419N | Cryptography & Information Security | CSE-425N | Expert Systems |
|----------|-------------------------------------|----------|----------------|

Note:

***The students will choose any two departmental electives courses out of the given elective list in 7thSemester.**

****Project should be initiated in the beginning of 7thsemester, and should be completed by the end of 8thsemester with good Report and power-point Presentation etc.**

*****4-6 weeks hand on training completed after 6thSemester Exams.**

INTERNATIONAL INSTITUTE OF ENGINEERING & TECHNOLOGY

| CSE-401N | Unix & Linux Programming | | | | | |
|-----------------------------|--|-----------|------------|------------|-------|--------|
| Lecture | Tutorial | Practical | Major Test | Minor Test | Total | Time |
| 4 | 0 | 0 | 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.

INTERNATIONAL INSTITUTE OF ENGINEERING & TECHNOLOGY

| CSE-403N Computer Graphics and Animation | | | | | | |
|---|---|------------------|-------------------|-------------------|--------------|----------------|
| Lecture | Tutorial | Practical | Major Test | Minor Test | Total | Time |
| 4 | 0 | 0 | 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-I

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

Unit-II

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-III

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-IV

Representation of 3-D Curves and Surfaces: interpolation and approximation alines, 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.

Reference Book

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

INTERNATIONAL INSTITUTE OF ENGINEERING & TECHNOLOGY

| CSE-415N | Object Oriented Software Engineering | | | | | |
|-----------------------------|---|-----------|---------------|------------|-------|--------|
| Lecture | Tutorial 1 | Practical | Major Test | Minor Test | Total | Time |
| 3 | 0 | 0 | 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.

| CSE-417N | Cyber Security | | | | | |
|------------------------|--|-----------|------------|------------|-------|--------|
| Lecture | Tutorial | Practical | Major Test | Minor Test | Total | Time |
| 3 | 0 | 0 | 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. | | | | | |

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, Magnus 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.

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-II

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-III

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-IV

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.

INTERNATIONAL INSTITUTE OF ENGINEERING & TECHNOLOGY

| CSE-419N | | | | | | |
|--|---|------------------|-------------------|-------------------|--------------|---------------|
| Cryptography and Information Security | | | | | | |
| Lecture | Tutorial | Practical | Major Test | Minor Test | Total | Time |
| 3 | 0 | 0 | 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 I: 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 II: 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 III: 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 IV: 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. Oded Goldreich. 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 Stallings, 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-421N | Agile Software Engineering | | | | | |
|-----------------------------|---|-----------|------------|------------|-------|--------|
| Lecture | Tutorial | Practical | Major Test | Minor Test | Total | Time |
| 3 | 0 | 0 | 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.

INTERNATIONAL INSTITUTE OF ENGINEERING & TECHNOLOGY

| CSE-423N | Big Data and Analytics | | | | | |
|----------------------|--|-----------|------------|------------|-------|--------|
| Lecture | Tutorial | Practical | Major Test | Minor Test | Total | Time |
| 3 | 0 | 0 | 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-425N | Expert Systems | | | | | |
|----------------------------|--|-----------|------------|------------|-------|--------|
| Lecture | Tutorial | Practical | Major Test | Minor Test | Total | Time |
| 3 | 0 | 0 | 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-I

Introduction to AI programming languages, Blind search strategies, Breadth first – Depth first – Heuristic search techniques Hill Climbing – Best first – A Algorithms AO* algorithm – game trees, 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-II

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-III

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

Unit-IV

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. Stuart 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.

INTERNATIONAL INSTITUTE OF ENGINEERING & TECHNOLOGY

| CSE-405N | Computer Graphics Lab | | | | | |
|-----------------------------|--|-----------|------------|-----------|-------|-------|
| Lecture | Tutorial | Practical | Minor Test | Practical | Total | Time |
| 0 | 1 | 2 | 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 using 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.

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

| CSE-409N | Unix & Linux Programming Lab | | | | | |
|-----------------------------|---|-----------|------------|-----------|-------|--------|
| Lecture | Tutorial | Practical | Minor Test | Practical | Total | Time |
| 0 | 0 | 2 | 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 operating systems 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.