



ACE Engineering College

An Autonomous Institution

(NBA ACCREDITED B.TECH COURSES, ACCORDED NAAC 'A' GRADE)

Ghatkesar, Hyderabad- 501 301

R22 - B.TECH. FOUR YEAR DEGREE COURSE **DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING** **COURSE STRUCTURE**

IV Year I Semester – Regulation R22

S.No.	Course Code	Course Title	Periods Per Week			Credits
			L	T	P	
1	CS701PC	Cryptography and Network Security	3	0	0	3
2	CS702PC	Compiler Design	3	0	0	3
		Professional Elective – IV				
3	CS741PE	Graph Theory	3	0	0	3
	CS742PE	Cyber Security				
	CS743PE	Soft Computing				
	CS744PE	Cloud Computing				
	CS745PE	Ad hoc & Sensor Networks				
		Professional Elective – V				
4	CS751PE	Advanced Algorithms	3	0	0	3
	CS752PE	Agile Methodology				
	CS753PE	Robotic Process Automation				
	CS754PE	Blockchain Technology				
	CS755PE	Software Process & Project Management				
5		Open Elective – II	3	0	0	3
6	CS703PC	Cryptography and Network Security Lab	0	0	2	1
7	CS704PC	Compiler Design Lab	0	0	2	1
8	CS705PC	Project Stage - I	0	0	6	3
		Total Credits	15	0	10	20

CS701PC: CRYPTOGRAPHY AND NETWORK SECURITY

B.Tech. IV Year I Sem.

L T P C
3 0 0 3

Course Objectives:

- Explain the importance and application of each of confidentiality, integrity, authentication and availability
- Understand various cryptographic algorithms.
- Understand the basic categories of threats to computers and networks
- Describe public-key cryptosystem.
- Describe the enhancements made to IPv4 by IPSec
- Understand Intrusions and intrusion detection

Course Outcomes:

- Student will be able to understand basic cryptographic algorithms, message and web authentication and security issues.
- Ability to identify information system requirements for both of them such as client and server.
- Ability to understand the current legal issues towards information security.

UNIT - I

Security Concepts: Introduction, The need for security, Security approaches, Principles of security, Types of Security attacks, Security services, Security Mechanisms, A model for Network Security

Cryptography Concepts and Techniques: Introduction, plain text and cipher text, substitution techniques, transposition techniques, encryption and decryption, symmetric and asymmetric key cryptography, steganography, key range and key size, possible types of attacks.

UNIT - II

Symmetric key Ciphers: Block Cipher principles, DES, AES, Blowfish, RC5, IDEA, Block cipher operation, Stream ciphers, RC4.

Asymmetric key Ciphers: Principles of public key cryptosystems, RSA algorithm, Elgamal Cryptography, Diffie-Hellman Key Exchange, Knapsack Algorithm.

UNIT - III

Cryptographic Hash Functions: Message Authentication, Secure Hash Algorithm (SHA-512),

Message authentication codes: Authentication requirements, HMAC, CMAC, Digital signatures, Elgamal Digital Signature Scheme.

Key Management and Distribution: Symmetric Key Distribution Using Symmetric & Asymmetric Encryption, Distribution of Public Keys, Kerberos, X.509 Authentication Service, Public – Key Infrastructure

UNIT - IV

Transport-level Security: Web security considerations, Secure Socket Layer and Transport Layer Security, HTTPS, Secure Shell (SSH)

Wireless Network Security: Wireless Security, Mobile Device Security, IEEE 802.11 Wireless LAN, IEEE 802.11i Wireless LAN Security

UNIT - V

E-Mail Security: Pretty Good Privacy, S/MIME IP Security: IP Security overview, IP Security architecture, Authentication Header, Encapsulating security payload, Combining security associations, Internet Key Exchange

Case Studies on Cryptography and security: Secure Multiparty Calculation, Virtual Elections, Single sign On, Secure Inter-branch Payment Transactions, Cross site Scripting Vulnerability.

TEXT BOOKS:

1. Cryptography and Network Security - Principles and Practice: William Stallings, Pearson Education, 6th Edition
2. Cryptography and Network Security: Atul Kahate, Mc Graw Hill, 3rd Edition

REFERENCE BOOKS:

1. Cryptography and Network Security: C K Shyamala, N Harini, Dr T R Padmanabhan, Wiley India, 1st Edition.
2. Cryptography and Network Security: Forouzan Mukhopadhyay, Mc Graw Hill, 3rd Edition
3. Information Security, Principles, and Practice: Mark Stamp, Wiley India.
4. Principles of Computer Security: WM. Arthur Conklin, Greg White, TMH
5. Introduction to Network Security: Neal Krawetz, CENGAGE Learning
6. Network Security and Cryptography: Bernard Menezes, CENGAGE Learning

CS702PC: COMPILER DESIGN

B.Tech. IV Year I Sem.

L T P C
3 0 0 3

Prerequisites

1. A course on "Formal Languages and Automata Theory".
2. A course on "Computer Organization and architecture".
3. A course on "Data Structures".

Course Objectives:

- Introduce the major concepts of language translation and compiler design and impart the
- knowledge of practical skills necessary for constructing a compiler.
- Topics include phases of compiler, parsing, syntax directed translation, type checking use of symbol tables, code optimization techniques, intermediate code generation, code generation and data flow analysis.

Course Outcomes:

- Demonstrate the ability to design a compiler given a set of language features.
- Demonstrate the knowledge of patterns, tokens & regular expressions for lexical analysis.
- Acquire skills in using lex tool & yacc tool for developing a scanner and parser.
- Design and implement LL and LR parsers
- Design algorithms to do code optimization in order to improve the performance of a program in terms of space and time complexity.
- Design algorithms to generate machine code.

UNIT - I

Introduction: The structure of a compiler, the science of building a compiler, programming language basics

Lexical Analysis: The Role of the Lexical Analyzer, Input Buffering, Recognition of Tokens, The Lexical-Analyzer Generator Lex, Finite Automata, From Regular Expressions to Automata, Design of a Lexical-Analyzer Generator, Optimization of DFA-Based Pattern Matchers.

UNIT - II

Syntax Analysis: Introduction, Context-Free Grammars, Writing a Grammar, Top-Down Parsing, Bottom-Up Parsing, Introduction to LR Parsing: Simple LR, More Powerful LR Parsers, Using Ambiguous Grammars and Parser Generators.

UNIT - III

Syntax-Directed Translation: Syntax-Directed Definitions, Evaluation Orders for SDD's, Applications of Syntax-Directed Translation, Syntax-Directed Translation Schemes, Implementing L-Attributed SDD's.

Intermediate-Code Generation: Variants of Syntax Trees, Three-Address Code, Types and Declarations, Type Checking, Control Flow, Switch-Statements, Intermediate Code for Procedures.

UNIT - IV

Run-Time Environments: Stack Allocation of Space, Access to Nonlocal Data on the Stack, Heap Management, Introduction to Garbage Collection, Introduction to Trace-Based Collection.

Code Generation: Issues in the Design of a Code Generator, The Target Language, Addresses in the Target Code, Basic Blocks and Flow Graphs, Optimization of Basic Blocks, A Simple Code Generator, Peephole Optimization, Register Allocation and Assignment, Dynamic Programming Code-Generation

UNIT - V

Machine-Independent Optimization: The Principal Sources of Optimization, Introduction to Data-Flow Analysis, Foundations of Data-Flow Analysis, Constant Propagation, Partial-Redundancy Elimination, Loops in Flow Graphs.

TEXT BOOK:

1. Compilers: Principles, Techniques and Tools, Second Edition, Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman.

REFERENCE BOOKS:

1. Lex & Yacc - John R. Levine, Tony Mason, Doug Brown, O'reilly
2. Compiler Construction, Loudon, Thomson.

CS741PE: GRAPH THEORY (Professional Elective – IV)

B.Tech. IV Year I Sem.

L T P C
3 0 0 3

Course Objectives:

- Understanding graphs, trees, connected paths, applications of trees and graphs.

Course Outcomes:

- Know some important classes of graph theoretic problems;
- Prove central theorems about trees, matching, connectivity, coloring and planar graphs;
- Describe and apply some basic algorithms for graphs;
- Use graph theory as a modeling tool.

UNIT - I

Introduction-Discovery of graphs, Definitions, Subgraphs, Isomorphic graphs, Matrix representations of graphs, Degree of a vertex, Directed walks, paths and cycles, Connectivity in digraphs, Eulerian and Hamilton digraphs, Eulerian digraphs, Hamilton digraphs, Special graphs, Complements, Larger graphs from smaller graphs, Union, Sum, Cartesian Product, Composition, Graphic sequences, Graph theoretic model of the LAN problem, Havel-Hakimi criterion, Realization of a graphic sequence.

UNIT - II

Connected graphs and shortest paths - Walks, trails, paths, cycles, Connected graphs, Distance, Cut-vertices and cut-edges, Blocks, Connectivity, Weighted graphs and shortest paths, Weighted graphs, Dijkstra's shortest path algorithm, Floyd-Warshall shortest path algorithm.

UNIT - III

Trees- Definitions and characterizations, Number of trees, Cayley's formula, Kirchoff-matrix-tree theorem, Minimum spanning trees, Kruskal's algorithm, Prim's algorithm, Special classes of graphs, Bipartite Graphs, Line Graphs, Chordal Graphs, Eulerian Graphs, Fleury's algorithm, Chinese Postman problem, Hamilton Graphs, Introduction, Necessary conditions and sufficient conditions.

UNIT - IV

Independent sets coverings and matchings- Introduction, Independent sets and coverings: basic equations, Matchings in bipartite graphs, Hall's Theorem, Konig's Theorem, Perfect matchings in graphs, Greedy and approximation algorithms.

UNIT - V

Vertex Colorings- Basic definitions, Cliques and chromatic number, Mycielski's theorem, Greedy coloring algorithm, Coloring of chordal graphs, Brooks theorem, Edge Colorings, Introduction and Basics, Gupta-Vizing theorem, Class-1 and Class-2 graphs, Edge-coloring of bipartite graphs, Class-2 graphs, Hajos union and Class-2 graphs, A scheduling problem and equitable edge-coloring.

TEXT BOOKS:

1. J. A. Bondy and U. S. R. Murty. Graph Theory, volume 244 of Graduate Texts in Mathematics. Springer, 1st edition, 2008.
2. J. A. Bondy and U. S. R. Murty. Graph Theory with Applications.

REFERENCE BOOKS:

1. Lecture Videos: <http://nptel.ac.in/courses/111106050/13>
2. Introduction to Graph Theory, Douglas B. West, Pearson.

CS742PE: CYBER SECURITY (Professional Elective – IV)

B.Tech. IV Year I Sem.

L T P C
3 0 0 3

Course objectives:

- To understand various types of cyber-attacks and cyber-crimes.
- To learn threats and risks within the context of cyber security.
- To have an overview of the cyber laws & concepts of cyber forensics.
- To study the defensive techniques against these attacks.

Course Outcomes:

1. Analyze and evaluate the cyber security needs of an organization.
2. Understand Cyber Security Regulations and Roles of International Law.
3. Design and develop security architecture for an organization.
4. Understand fundamental concepts of data privacy attacks.

UNIT- I

Introduction to Cyber Security: Basic Cyber Security Concepts, layers of security, Vulnerability, threat, Harmful acts, Internet Governance – Challenges and Constraints, Computer Criminals, CIA Triad, Assets and Threat, motive of attackers, active attacks, passive attacks, Software attacks, hardware attacks, Cyber Threats-Cyber Warfare, Cyber Crime, Cyber terrorism, Cyber Espionage, etc., Comprehensive Cyber Security Policy.

UNIT - II

Cyberspace and the Law & Cyber Forensics: Introduction, Cyber Security Regulations, Roles of International Law. The INDIAN Cyberspace, National Cyber Security Policy. Introduction, Historical background of Cyber forensics, Digital Forensics Science, The Need for Computer Forensics, Cyber Forensics and Digital evidence, Forensics Analysis of Email, Digital Forensics Lifecycle, Forensics Investigation, Challenges in Computer Forensics.

UNIT - III

Cybercrime: Mobile and Wireless Devices: Introduction, Proliferation of Mobile and Wireless Devices, Trends in Mobility, Credit card Frauds in Mobile and Wireless Computing Era, Security Challenges Posed by Mobile Devices, Registry Settings for Mobile Devices, Authentication service Security, Attacks on Mobile/Cell Phones, Organizational security Policies and Measures in Mobile Computing Era, Laptops.

UNIT- IV

Cyber Security: Organizational Implications: Introduction, cost of cybercrimes and IPR issues, web threats for organizations, security and privacy implications, social media marketing: security risks and perils for organizations, social computing and the associated challenges for organizations.

UNIT - V

Privacy Issues: Basic Data Privacy Concepts: Fundamental Concepts, Data Privacy Attacks, Data linking and profiling, privacy policies and their specifications, privacy policy languages, privacy in different domains- medical, financial, etc.

Cybercrime: Examples and Mini-Cases

Examples: Official Website of Maharashtra Government Hacked, Indian Banks Lose Millions of Rupees, Parliament Attack, Pune City Police Bust Nigerian Racket, e-mail spoofing instances.

Mini-Cases: The Indian Case of online Gambling, An Indian Case of Intellectual Property Crime, Financial Frauds in Cyber Domain.

TEXT BOOKS:

1. Nina Godbole and Sunit Belpure, Cyber Security Understanding Cyber Crimes, Computer Forensics and Legal Perspectives, Wiley

REFERENCE BOOKS:

1. B. B. Gupta, D.P. Agrawal, Haoxiang Wang, Computer and Cyber Security: Principles, Algorithm, Applications, and Perspectives, CRC Press
2. Cyber Security Essentials, James Graham, Richard Howard and Ryan Otson, CRC Press.
3. Introduction to Cyber Security, Chwan-Hwa(john) Wu, J.David Irwin, CRC Press T&F Group.

CS743PE: SOFT COMPUTING (Professional Elective – IV)

B.Tech. IV Year I Sem.

L T P C
3 0 0 3

Course Objectives:

- Familiarize with soft computing concepts
- Introduce and use the idea of fuzzy logic and use of heuristics based on human experience
- Familiarize the Neuro-Fuzzy modeling using Classification and Clustering techniques
- Learn the concepts of Genetic algorithm and its applications
- Acquire the knowledge of Rough Sets.

Course Outcomes:

- Identify the difference between Conventional Artificial Intelligence to Computational Intelligence.
- Understand fuzzy logic and reasoning to handle and solve engineering problems
- Apply the Classification techniques on various applications.
- Perform various operations of genetic algorithms and Rough Sets.

UNIT - I

Introduction to Soft Computing: Evolutionary Computing, "Soft" computing versus "Hard" computing, Soft Computing Methods, Recent Trends in Soft Computing, Characteristics of Soft computing, Applications of Soft Computing Techniques.

UNIT- II

Fuzzy Systems: Fuzzy Sets, Fuzzy Relations, Fuzzy Logic, Fuzzy Rule-Based Systems

UNIT- III

Fuzzy Decision Making, Particle Swarm Optimization

UNIT- IV

Genetic Algorithms: Basic Concepts, Basic Operators for Genetic Algorithms, Crossover and Mutation Properties, Genetic Algorithm Cycle, Fitness Function, Applications of Genetic Algorithm.

UNIT- V

Rough Sets, Rule Induction, and Discernibility Matrix, Integration of Soft Computing Techniques.

TEXT BOOK:

1. Soft Computing - Advances and Applications - Jan 2015 by B.K. Tripathy and J. Anuradha - Cengage Learning

REFERENCE BOOKS:

1. S. N. Sivanandam & S. N. Deepa, "Principles of Soft Computing", 2nd edition, Wiley India, 2008.
2. David E. Goldberg, "Genetic Algorithms-In Search, optimization and Machine learning", Pearson Education.
3. J. S. R. Jang, C.T. Sun and E.Mizutani, "Neuro-Fuzzy and Soft Computing", Pearson Education, 2004.
4. G.J. Klir & B. Yuan, "Fuzzy Sets & Fuzzy Logic", PHI, 1995.
5. Melanie Mitchell, "An Introduction to Genetic Algorithm", PHI, 1998.
6. Timothy J. Ross, "Fuzzy Logic with Engineering Applications", McGraw- Hill International editions, 1995

CS744PE: CLOUD COMPUTING (Professional Elective – IV)

B.Tech. IV Year I Sem.

L T P C
3 0 0 3

Pre-requisites:

1. A course on "Computer Networks".
2. A course on "Operating System".

Course Objectives:

- This course provides an insight into cloud computing
- Topics covered include- Cloud Computing Architecture, Deployment Models, Service Models, Technological Drivers for Cloud Computing, Networking for Cloud Computing and Security in Cloud Computing

Course Outcomes:

- Understand different computing paradigms and potential of the paradigms and specifically cloud computing
- Understand cloud service types, cloud deployment models and technologies supporting and driving the cloud
- Acquire the knowledge of programming models for cloud and development of software application that runs the cloud and various services available from major cloud providers
- Understand the security concerns and issues in cloud computing
- Acquire the knowledge of advances in cloud computing.

UNIT - I

Computing Paradigms, Cloud Computing Fundamentals, Cloud Computing Architecture and Management

UNIT - II

Cloud Deployment Models, Cloud Service Models, Technological Drivers for Cloud Computing: SOA and Cloud, Multicore Technology, Web 2.0 and Web 3.0, Pervasive Computing, Operating System, Application Environment

UNIT - III

Virtualization, Programming Models for Cloud Computing: MapReduce, Cloud Haskell, Software Development in Cloud

UNIT - IV

Networking for Cloud Computing: Introduction, Overview of Data Center Environment, Networking Issues in Data Centers, Transport Layer Issues in DCNs, Cloud Service Providers

UNIT - V

Security in Cloud Computing, and Advanced Concepts in Cloud Computing

TEXT BOOK:

1. Chandrasekaran, K. *Essentials of cloud computing*. CRC Press, 2014

REFERENCE BOOKS:

1. Cloud Computing: Principles and Paradigms, Editors: Rajkumar Buyya, James Broberg, Andrzej M. Goscinski, Wiley, 2011
2. Enterprise Cloud Computing - Technology, Architecture, Applications, Gautam Shroff, Cambridge University Press, 2010
3. Cloud Computing Bible, Barrie Sosinsky, Wiley-India, 2010

CS745PE: AD-HOC & SENSOR NETWORKS (Professional Elective – IV)

B.Tech. IV Year I Sem.

L T P C
3 0 0 3

Prerequisites

1. Computer Networks
2. Distributed Systems
3. Mobile Computing

Course Objectives

- To understand the challenges of routing in ad-hoc and sensor networks
- To understand various broadcast, multicast and geocasting protocols in ad hoc and sensor networks
- To understand basics of Wireless sensors, and Lower Layer Issues and Upper Layer Issues of WSN

Course Outcomes

- Understand the concepts of sensor networks and applications
- Understand and compare the MAC and routing protocols for adhoc networks
- Understand the transport protocols of sensor networks

UNIT - I

Introduction to Ad Hoc Networks

Characteristics of MANETs, Applications of MANETs and Challenges of MANETs.

Routing in MANETs

Criteria for classification, Taxonomy of MANET routing algorithms, *Topology-based* routing algorithms- Proactive: DSDV, WRP; Reactive: DSR, AODV, TORA; Hybrid: ZRP; *Position-based* routing algorithms- Location Services-DREAM, Quorum-based, GLS; Forwarding Strategies, Greedy Packet, Restricted Directional Flooding-DREAM, LAR; Other routing algorithms-QoS Routing, CEDAR.

UNIT - II

Data Transmission

Broadcast Storm Problem, Rebroadcasting Schemes-Simple-flooding, Probability-based Methods, Area-based Methods, Neighbour Knowledge-based: SBA, Multipoint Relaying, AHBP. Multicasting: Tree-based: AMRIS, MAODV; Mesh-based: ODMRP, CAMP; Hybrid: AMRoute, MCEDAR.

UNIT - III

Geocasting

Data-transmission Oriented-LBM; Route Creation Oriented-GeoTORA, MGR.

TCP over Ad Hoc TCP protocol overview, TCP and MANETs, Solutions for TCP over Ad hoc

UNIT - IV

Basics of Wireless Sensors and Lower Layer Issues-Applications, Classification of sensor networks, Architecture of sensor network, Physical layer, MAC layer, Link layer, Routing Layer.

UNIT - V

Upper Layer Issues of WSN

Transport layer, High-level application layer support, Adapting to the inherent dynamic nature of WSNs.

TEXT BOOKS

1. Ad Hoc and Sensor Networks - Theory and Applications, *Carlos Corderio Dharma P. Aggarwal*, World Scientific Publications, March 2006, ISBN - 981-256-681-3
2. Wireless Sensor Networks: An Information Processing Approach, Feng Zhao, Leonidas Guibas, Elsevier Science, ISBN - 978-1-55860-914-3 (Morgan Kauffman)

REFERENCE BOOKS:

1. C. Siva Ram Murthy, B.S. Manoj Ad Hoc Wireless Networks: Architectures and Protocols.
2. Taieb Znati Kazem Sohraby, Daniel Minoli, Wireless Sensor Networks: Technology, Protocols and Applications, Wiley.

CS751PE: ADVANCED ALGORITHMS (Professional Elective – V)

B.Tech. IV Year I Sem.

L T P C
3 0 0 3

Pre-Requisites: Algorithm Design and Analysis

Course Objectives:

- To familiarize advanced methods on analysis of algorithms.
- To familiarize with graphs and algorithms related shortest path
- To understand matrix computations and modulo representations
- To introduce randomized, approximation algorithms and computational complexity topics

Course Outcomes:

- Familiarize with advanced methods on analysis of algorithms
- Familiarize with the graphs, graph matching and shortest path algorithms
- Understand matrix computations and modulo representations
- Understand randomized, approximation algorithms and computational complexity topics

UNIT – I

Introduction to Algorithms, Classification of Algorithms, Asymptotic Analysis, Introduction to Recurrence equations - Linear recurrences, Non-linear recurrences, Formulation of recurrence equations, techniques for solving recurrence equations, Solving recurrence equations using polynomial reduction, Master's theorem

Graph: Definitions and Elementary Algorithms: Shortest path by BFS, shortest path in edge-weighted case (Dijkstra's), depth-first search and computation of strongly connected components, Multistage Graph, topological sorting

UNIT – II

Graph Matching: Algorithm to compute maximum matching. Characterization of maximum matching by augmenting paths, Edmond's Blossom algorithm to compute augmenting path, Bipartite matching problem

Matroids: Introduction to greedy paradigm, algorithm to compute a maximum weight maximal independent set, Optimal tree problems- optimal merge, Huffman coding, tree vertex splitting problem.

Shortest Path in Graphs: Floyd-Warshall algorithm, Travelling Sales Person Problem and introduction to dynamic programming paradigm. Optimal Graph Problems - Minimum Spanning Tree, Single source shortest path.

UNIT - III

Flow-Networks: Maxflow - mincut theorem, Ford-Fulkerson Method to compute maximum flow, Edmond-Karp maximum-flow algorithm.

Matrix Computations: Strassen's algorithm and introduction to divide and conquer paradigm, Chain Matrix Multiplication, Matrix operations - Gaussian Elimination method, LUP-decomposition, Crout's method of decomposition, inverse of a triangular matrix,

UNIT - IV

Modulo Representation of integers/polynomials: Chinese Remainder Theorem, Conversion between base-representation and modulo-representation, interpolation problem. Multiplication of long integers by using Divide and Conquer paradigm, Schonhage-Strassen's Integer Multiplication algorithm.

String Algorithms: Naïve String, Rabin Karp, KMP, Boyer Moore, Harspool algorithms

UNIT - V

Basics of Computational Complexity: Introduction to computational complexity, complexity classes, Satisfiability problem and Cook's theorem, Examples of NP- Complete problems

Randomized algorithms: Introduction, Types of Randomized algorithms, Example of Randomized algorithms.

Approximation algorithms: Introduction, Types of Approximation algorithms, Examples of Approximation algorithms

TEXT BOOK:

1. Design and Analysis of Algorithms, S. Sridhar, Oxford University Press.

REFERENCE BOOKS:

1. Introduction to Algorithms, Cormen, Leiserson, Rivest, Stein.
2. The Design and Analysis of Computer Algorithms, Aho, Hopcroft, Ullman.
3. Algorithm Design, Kleinberg and Tardos.

CS752PE: AGILE METHODOLOGY (Professional Elective – V)

B.Tech. IV Year I Sem.

L T P C
3 0 0 3

Course Objectives:

- Knowledge on concepts of agile development, releasing, planning and developing

Course Outcomes:

- Identify basic concepts of agile methodology and Extreme programming
- Analyze real customer involvement in collaboration
- Discuss risk management and iteration planning
- Understanding incremental requirements, refactoring, incremental design and architecture

UNIT - I

Introduction Extreme Programming (XP) - Agile Development

Why Agile?, Understanding Success, Beyond Deadlines, Importance of Organizational Success, Introduction to Agility, Agile methods-Scrum and XP, Manifesto for Agile Software Development, Principles of Agile Process. Understanding XP (Extreme Programming) - XP life cycle, XP team, XP Concepts, Adopting XP - Knowing whether XP is suitable, Implementing XP, assessing Agility, Practicing XP - Thinking, Pair Programming, Energized work, Informative Workspace, Root cause Analysis, Retrospectives.

UNIT - II

Collaborating

Trust, Sit together, Real customer involvement, Ubiquitous language, Stand-Up meetings, coding standards, Iteration demo, Reporting.

UNIT - III

Releasing

Bugfree Release, Version Control, Ten-Minute Build, continuous integration, Collective ownership and Documentation.

UNIT – IV

Planning

Version, Release Planning, The Planning Game, Risk Management, Iteration Planning, Slack, Stories, and Estimating

UNIT - V

Developing

Incremental requirements, Customer tests, Test driven development, Refactoring, Incremental design and architecture, spike solutions, Performance optimization, Exploratory testing.

TEXT BOOK:

1. The art of Agile Development, James Shore and Shane Warden, 11th Indian Reprint, O'Reilly, 2018.

REFERENCE BOOKS:

1. Learning Agile, Andrew Stellman and Jennifer Greene, O'Reilly, 4th Indian Reprint, 2018
2. Practices of an Agile Developer, Venkat Subramaniam and Andy Hunt, SPD, 5th Indian Reprint, 2015
3. Agile Project Management - Jim Highsmith, Pearson Low price Edition 2004

CS753PE: ROBOTIC PROCESS AUTOMATION (Professional Elective – V)

B.Tech. IV Year I Sem.

L T P C
3 0 0 3

Course Objectives:

- Introduce robotic process automation, techniques of automation using UiPath RPA tool.

Course Outcomes:

- Understand the concepts of Robotic Process Automation.
- Apply the flow chart mechanism in various calculations.
- Applying UiPath tool for debugging process
- Design system managing techniques.
- Create application for process automation using UiPath tool.

UNIT - I

Robotic Process Automation: Introduction, Scope and techniques of automation, Robotic process automation, Components of RPA, RPA platforms, About UiPath

UiPath Stack UiPath Studio, UiPath Robot, Types of Robots, UiPath Orchestrator

UiPath Studio Projects, User interface

The User Interface: Task recorder, Advanced UI interactions: Input methods, Output methods

UNIT - II

Sequence, Flowchart, and Control Flow: Sequencing the workflow, Activities, Control Flow, various types of loops and decision making

Data Manipulation: Variables and scope, Collections, Arguments - Purpose and use, Data table usage with examples, File operation with step-by-step example, CSV/Excel to data table and vice versa

UNIT - III

Taking Control of the Controls: Finding and attaching windows, Finding the control, Techniques for waiting for a control, Act on controls - mouse and keyboard activities, Handling events, revisit recorder, When to use OCR, Types of OCR available, How to use OCR

Plugins and Extensions: Terminal Plugin, SAP Automation, Citrix automation and Credential management

UNIT - IV

Handling User Events and Assistant Bots: Assistant bots, Monitoring system event triggers, Monitoring image and element triggers, Launching an assistant bot on a keyboard event

Exception Handling, Debugging, and Logging: Exception handling, Common exceptions and ways to handle them, Logging and taking screenshots, Debugging techniques, Collecting crash dumps, Error reporting

UNIT - V

Managing and Maintaining the Code: Project organization, nesting workflows, Reusability of workflows, Commenting techniques, State Machine, When to use Flowcharts, State Machines, or Sequences, Using config files

Deploying and Maintaining the Bot: Publishing using publish utility, using Orchestration Server to control bots, deploy bots, License Management, Publishing and Managing updates

TEXT BOOK:

1. Learning Robotic Process Automation: Create Software robots and automate business processes with the leading RPA tool - UiPath: Create Software robots. with the leading RPA tool – UiPath Kindle Edition

REFERENCE BOOK:

1. Robotic Process Automation A Complete Guide - 2020 Edition Kindle Edition.

CS754PE: BLOCKCHAIN TECHNOLOGY (Professional Elective – V)

B.Tech. IV Year I Sem.

L T P C
3 0 0 3

Prerequisites:

1. Knowledge in information security and applied cryptography.
2. Knowledge in Computer Networks

Course Objectives:

- To learn the fundamentals of Blockchain and various types of block chain and consensus mechanisms.
- To understand the public block chain system, Private block chain system and consortium blockchain.
- Able to know the security issues of blockchain technology.

Course Outcomes:

- Understanding concepts behind crypto currency
- Applications of smart contracts in decentralized application development
- Understand frameworks related to public, private and hybrid blockchain
- Create blockchain for different application case studies

UNIT-I

Fundamentals of Blockchain: Introduction, Origin of Blockchain, Blockchain Solution, Components of Blockchain, Block in a Blockchain, The Technology and the Future.

Blockchain Types and Consensus Mechanism: Introduction, Decentralization and Distribution, Types of Blockchain, Consensus Protocol.

Cryptocurrency – Bitcoin, Altcoin and Token: Introduction, Bitcoin and the Cryptocurrency, Cryptocurrency Basics, Types of Cryptocurrencies, Cryptocurrency Usage.

UNIT-II

Public Blockchain System: Introduction, Public Blockchain, Popular Public Blockchains, The Bitcoin Blockchain, Ethereum Blockchain.

Smart Contracts: Introduction, Smart Contract, Characteristics of a Smart Contract, Types of Smart Contracts, Types of Oracles, Smart Contracts in Ethereum, Smart Contracts in Industry.

UNIT-III

Private Blockchain System: Introduction, Key Characteristics of Private Blockchain, Need of Private Blockchain, Private Blockchain Examples, Private Blockchain and Open Source, E-commerce Site Example, Various Commands (Instructions) in E-commerce Blockchain, Smart Contract in Private Environment, State Machine, Different Algorithms of Permissioned Blockchain, Byzantine Fault, Multichain.

Consortium Blockchain: Introduction, Key Characteristics of Consortium Blockchain, Need of Consortium Blockchain, Hyperledger Platform, Overview of Ripple, Overview of Corda.

Initial Coin Offering: Introduction, Blockchain Fundraising Methods, Launching an ICO, Investing in an ICO, Pros and Cons of Initial Coin Offering, Successful Initial Coin Offerings, Evolution of ICO, ICO Platforms.

UNIT-IV

Security in Blockchain: Introduction, Security Aspects in Bitcoin, Security and Privacy Challenges of Blockchain in General, Performance and Scalability, Identity Management and Authentication, Regulatory Compliance and Assurance, Safeguarding Blockchain Smart Contract (DApp), Security Aspects in Hyperledger Fabric.

Applications of Blockchain: Introduction, Blockchain in Banking and Finance, Blockchain in Education, Blockchain in Energy, Blockchain in Healthcare, Blockchain in Real-estate, Blockchain In Supply Chain, The Blockchain and IoT. Limitations and Challenges of Blockchain.

UNIT-V

Blockchain Case Studies: Case Study 1 - Retail, Case Study 2 - Banking and Financial Services, Case Study 3 - Healthcare, Case Study 4 - Energy and Utilities.

Blockchain Platform using Python: Introduction, Learn How to Use Python Online Editor, Basic Programming Using Python, Python Packages for Blockchain.

Blockchain platform using Hyperledger Fabric: Introduction, Components of Hyper ledger Fabric Network, Chain codes from Developer.ibm.com, Blockchain Application Using Fabric Java SDK.

TEXT BOOK:

1. "Blockchain Technology", Chandramouli Subramanian, Asha A. George, Abhilasj K A and Meena Karthikeyan, Universities Press.

REFERENCE BOOKS:

1. Michael Juntao Yuan, Building Blockchain Apps, Pearson, India.
2. Blockchain Blueprint for Economy, Melanie Swan, SPD O'reilly.
3. Blockchain for Business, Jai Singh Arun, Jerry Cuomo, Nitin Gaur, Pearson.

CS755PE: SOFTWARE PROCESS & PROJECT MANAGEMENT (Professional Elective – V)

B.Tech. IV Year I Sem.

L T P C
3 0 0 3

Course Objectives:

- To acquire knowledge on software process management.
- To acquire managerial skills for software project development.
- To understand software economics.

Course Outcomes:

- Understand the software process change, assessment, project plans and Quality Standards.
- Examine the life cycle phases, artifacts, workflows and checkpoints of a process.
- Design and develop software products using conventional and modern principles of software project management.
- Identify the new project management process and practices.

UNIT - I

Software Process Maturity

Software maturity Framework, Principles of Software Process Change, Software Process Assessment, The Initial Process, The Repeatable Process, The Defined Process, The Managed Process, The Optimizing Process, Process Reference Models Capability Maturity Model (CMM), CMMI, PCMM, PSP, TSP).

UNIT - II

Software Project Management Renaissance

Conventional Software Management, Evolution of Software Economics, Improving Software Economics, Life-Cycle Phases and Process artifacts
Engineering and Production stages, inception phase, elaboration phase, construction phase, transition phase, artifact sets, management artifacts, engineering artifacts and pragmatic artifacts, model-based software architectures.

UNIT - III

Workflows and Checkpoints of process

Software process workflows, Iteration workflows, Major milestones, minor milestones, periodic status assessments, Process Planning Work breakdown structures, Planning guidelines, cost and schedule estimating process, iteration planning process, Pragmatic planning.

UNIT - IV

Project Organizations

Line-of- business organizations, project organizations, evolution of organizations, process automation. Project Control and process instrumentation, The seven-core metrics, management indicators, quality indicators, life-cycle expectations, Pragmatic software metrics, metrics automation.

UNIT - V

CCPDS-R Case Study and Future Software Project Management Practices, Modern Project Profiles, Next-Generation software Economics, Modern Process Transitions.

TEXT BOOKS:

1. Managing the Software Process, Watts S. Humphrey, Pearson Education
2. Software Project Management, Walker Royce, Pearson Education

REFERENCE BOOKS:

1. An Introduction to the Team Software Process, Watts S. Humphrey, Pearson Education, 2000

2. Process Improvement essentials, James R. Persse, O'Reilly, 2006
3. Software Project Management, Bob Hughes & Mike Cotterell, fourth edition, TMH, 2006
4. Applied Software Project Management, Andrew Stellman & Jennifer Greene, O'Reilly, 2006.
5. Software Engineering Project Management, Richard H. Thayer & Edward Yourdon, 2nd edition, Wiley India, 2004.
6. Agile Project Management, Jim Highsmith, Pearson education, 2004.

CS7210E: OPERATING SYSTEMS (Open Elective –II)

B.Tech. IV Year I Sem.

L T P C
3 0 0 3

Prerequisites:

1. A course on “Computer Programming and Data Structures”.
2. A course on “Computer Organization and Architecture”.

Course Objectives:

- Introduce operating system concepts (i.e., processes, threads, scheduling, synchronization, deadlocks, memory management, file and I/O subsystems and protection)
- Introduce the issues to be considered in the design and development of operating system
- Introduce basic Unix commands, system call interface for process management, interprocess communication and I/O in Unix

Course Outcomes:

- Will be able to control access to a computer and the files that may be shared
- Demonstrate the knowledge of the components of computers and their respective roles in computing.
- Ability to recognize and resolve user problems with standard operating environments.
- Gain practical knowledge of how programming languages, operating systems, and architectures interact and how to use each effectively.

UNIT - I

Operating System - Introduction, Structures - Simple Batch, Multiprogrammed, Time-shared, Personal Computer, Parallel, Distributed Systems, Real-Time Systems, System components, Operating System services, System Calls

Process - Process concepts and scheduling, Operations on processes, Cooperating Processes, Threads

UNIT - II

CPU Scheduling - Scheduling Criteria, Scheduling Algorithms, Multiple -Processor Scheduling. System call interface for process management-fork, exit, wait, waitpid, exec

Deadlocks - System Model, Deadlocks Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, and Recovery from Deadlock

UNIT - III

Process Management and Synchronization - The Critical Section Problem, Synchronization Hardware, Semaphores, and Classical Problems of Synchronization, Critical Regions, Monitors

Interprocess Communication Mechanisms: IPC between processes on a single computer system, IPC between processes on different systems, using pipes, FIFOs, message queues, shared memory.

UNIT - IV

Memory Management and Virtual Memory - Logical versus Physical Address Space, Swapping, Contiguous Allocation, Paging, Segmentation, Segmentation with Paging, Demand Paging, Page Replacement, Page Replacement Algorithms.

UNIT - V

File System Interface and Operations -Access methods, Directory Structure, Protection, File System Structure, Allocation methods, Free-space Management. Usage of open, create, read, write, close, lseek, stat, ioctl system calls.

TEXT BOOKS:

1. Operating System Principles- Abraham Silberchatz, Peter B. Galvin, Greg Gagne 7th Edition, John Wiley
2. Advanced programming in the UNIX environment, W.R. Stevens, Pearson education.

REFERENCE BOOKS:

1. Operating Systems- Internals and Design Principles, William Stallings, Fifth Edition-2005, Pearson Education/PHI
2. Operating System A Design Approach- Crowley, TMH.
3. Modern Operating Systems, Andrew S. Tanenbaum 2nd edition, Pearson/PHI
4. UNIX programming environment, Kernighan and Pike, PHI/ Pearson Education
5. UNIX Internals -The New Frontiers, U. Vahalia, Pearson Education.

CS722OE: SOFTWARE ENGINEERING (Open Elective –II)

B.Tech. IV Year I Sem.

L T P C
3 0 0 3

Course Objectives

- The aim of the course is to provide an understanding of the working knowledge of the techniques for estimation, design, testing and quality management of large software development projects.
- Topics include process models, software requirements, software design, software testing, software process/product metrics, risk management, quality management and UML diagrams

Course Outcomes

- Ability to translate end-user requirements into system and software requirements, using e.g.
- UML, and structure the requirements in a Software Requirements Document (SRD).
- Identify and apply appropriate software architectures and patterns to carry out high level design of a system and be able to critically compare alternative choices.
- Will have experience and/or awareness of testing problems and will be able to develop a simple testing report

UNIT - I

Introduction to Software Engineering: The evolving role of software, changing nature of software, software myths.

A Generic view of process: Software engineering- a layered technology, a process framework, the capability maturity model integration (CMMI)

Process models: The waterfall model, Spiral model and Agile methodology

UNIT - II

Software Requirements: Functional and non-functional requirements, user requirements, system requirements, interface specification, the software requirements document.

Requirements engineering process: Feasibility studies, requirements elicitation and analysis, requirements validation, requirements management.

UNIT - III

Design Engineering: Design process and design quality, design concepts, the design model.

Creating an architectural design: software architecture, data design, architectural styles and patterns, architectural design, conceptual model of UML, basic structural modeling, class diagrams, sequence diagrams, collaboration diagrams, use case diagrams, component diagrams.

UNIT - IV

Testing Strategies: A strategic approach to software testing, test strategies for conventional software, black-box and white-box testing, validation testing, system testing, the art of debugging.

Metrics for Process and Products: Software measurement, metrics for software quality.

UNIT - V

Risk management: Reactive Vs proactive risk strategies, software risks, risk identification, risk projection, risk refinement, RMMM

Quality Management: Quality concepts, software quality assurance, software reviews, formal technical reviews, statistical software quality assurance, software reliability, the ISO 9000 quality standards.

TEXT BOOKS:

1. Software Engineering, A practitioner's Approach- Roger S. Pressman, 6th edition, McGraw Hill International Edition.

2. Software Engineering- Sommerville, 7th edition, Pearson Education.

REFERENCE BOOKS:

1. The unified modeling language user guide Grady Booch, James Rumbaugh, Ivar Jacobson, Pearson Education.
2. Software Engineering, an Engineering approach- James F. Peters, Witold Pedrycz, John Wiley.
3. Software Engineering principles and practice- Waman S Jawadekar, The McGraw-Hill Companies.
4. Fundamentals of object-oriented design using UML Meiler page-Jones: Pearson Education.

CS703PC: CRYPTOGRAPHY AND NETWORK SECURITY LAB

B.Tech. IV Year I Sem.

L T P C
0 0 2 1

Course Objectives:

- Explain the objectives of information security
- Explain the importance and application of each of confidentiality, integrity, authentication and availability
- Understand various cryptographic algorithms.

Course Outcomes:

- Understand basic cryptographic algorithms, message and web authentication and security issues.
- Identify information system requirements for both of them such as client and server.
- Understand the current legal issues towards information security.

List of Experiments:

1. Write a C program that contains a string (char pointer) with a value 'Hello world'. The program should XOR each character in this string with 0 and display the result.
2. Write a C program that contains a string (char pointer) with a value 'Hello world'. The program should AND or and XOR each character in this string with 127 and display the result.
3. Write a Java program to perform encryption and decryption using the following algorithms
 - a. Ceaser cipher
 - b. Substitution cipher
 - c. Hill Cipher
4. Write a C/JAVA program to implement the DES algorithm logic.
5. Write a C/JAVA program to implement the Blowfish algorithm logic.
6. Write a C/JAVA program to implement the Rijndael algorithm logic.
7. Write the RC4 logic in Java Using Java cryptography; encrypt the text "Hello world" using Blowfish. Create your own key using Java key tool.
8. Write a Java program to implement the RSA algorithm.
9. Implement the Diffie-Hellman Key Exchange mechanism using HTML and JavaScript.
10. Calculate the message digest of a text using the SHA-1 algorithm in JAVA.
11. Calculate the message digest of a text using the MD5 algorithm in JAVA

TEXT BOOKS:

1. Cryptography and Network Security - Principles and Practice: William Stallings, Pearson Education, 6th Edition
2. Cryptography and Network Security: Atul Kahate, McGraw Hill, 3rd Edition

REFERENCE BOOKS:

1. Cryptography and Network Security: C K Shyamala, N Harini, Dr T R Padmanabhan, Wiley India, 1st Edition.
2. Cryptography and Network Security: Forouzan Mukhopadhyay, McGraw Hill, 3rd Edition
3. Information Security, Principles, and Practice: Mark Stamp, Wiley India.
4. Principles of Computer Security: WM. Arthur Conklin, Greg White, TMH
5. Introduction to Network Security: Neal Krawetz, CENGAGE Learning
6. Network Security and Cryptography: Bernard Menezes, CENGAGE Learning

CS704PC: COMPILER DESIGN LAB

B.Tech. IV Year I Sem.

L T P C
0 0 2 1

Prerequisites

1. A Course on "Object Oriented Programming through Java".

Co-requisites:

1. A course on "Web Technologies".

Course Objectives:

- To understand the various phases in the design of a compiler.
- To understand the design of top-down and bottom-up parsers.
- To understand syntax directed translation schemes.
- To introduce lex and yacc tools.

Course Outcomes:

- Design, develop, and implement a compiler for any language.
- Use lex and yacc tools for developing a scanner and a parser.
- Design and implement LL and LR parsers.

List of Experiments

1. Implementation of symbol table.
2. Develop a lexical analyzer to recognize a few patterns inc (ex. Identifiers, constants, comments, operators etc.)
3. Implementation of lexical analyzer using lex tool.
4. Generate yacc specification for a few syntactic categories.
 - a) Program to recognize a valid arithmetic expression that uses operator +, -, * and /.
 - b) Program to recognize a valid variable which starts with a letter followed by any number of letter or digits.
 - c) Implementation of calculator using lex and yacc.
5. Convert the bnf rules into yacc form and write code to generate abstract syntax tree.
6. Implement type checking
7. Implement any one storage allocation strategies (heap, stack, static)
8. Write a lex program to count the number of words and number of lines in a given file or program.
9. Write a 'C' program to implement lexical analyzer using c program.
10. write recursive descent parser for the grammar $E \rightarrow E+T$ $E \rightarrow T$ $T \rightarrow T^*F$ $T \rightarrow F$
 $F \rightarrow (E)/id$.
11. write recursive descent parser for the grammar $S \rightarrow (L)$ $S \rightarrow a$
 $L \rightarrow L,S$ $L \rightarrow S$
12. Write a C program to calculate first function for the grammar
 $E \rightarrow E+T$ $E \rightarrow T$ $T \rightarrow T^*F$ $T \rightarrow F$ $F \rightarrow (E)/id$
13. Write a YACC program to implement a top down parser for the given grammar.
13. Write a YACC program to evaluate algebraic expression.

TEXT BOOK:

1. Compilers: Principles, Techniques and Tools, Second Edition, Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman.

REFERENCE BOOKS:

1. Lex & Yacc - John R. Levine, Tony Mason, Doug Brown, O'reilly
2. Compiler Construction, Loudon, Thomson