Programme : Computer Science

Course Outcomes of the Course "Foundations of Computer Science"

CO1. Explain and manipulate the different concepts in automata theory and formal languages
   a. Such as formal proofs, (non-)deterministic automata, regular expressions, regular
   b. Languages, context-free grammars, context-free languages, Turing machines

CO2. Design machines of these types to carry out simple computational tasks.

CO3. Explain the power and the limitations of regular languages and context-free languages.

CO4. Prove properties of languages, grammars and automata with rigorously formal Mathematical methods.

CO5. Design automata, regular expressions and context-free grammars accepting or generating certain language.

CO6. Describe the language accepted by an automata or generated by a regular expression or a Context-free grammar.

CO7. Transform between equivalent deterministic and non-deterministic finite automata, and Regular expressions.

CO8. Simplify Automata and Context-Free Grammars.

CO9. Determine if a certain word belongs to a language.

CO10. Define Turing machines performing simple tasks.

Course Outcomes of the Course "Algorithmic"

CO1. Define basic data structures and relevant standard algorithms for them

CO2. Analyze worst-case running times of algorithms using asymptotic analysis.

CO3. Compare and implementation of various balanced search trees.

CO4. Describe the divide-and-conquer paradigm and explain when an algorithmic design situation calls for it.

CO5. Describe the greedy paradigm and explain when an algorithmic design situation calls for it

CO6. Explain the different ways to analyze randomized algorithms.

CO7. Describe the dynamic-programming paradigm and explain when an algorithmic design situation calls for it.

CO8. Explain the major graph algorithms and their analyses

CO9. Explain what an approximation algorithm is, and the benefit of using approximation algorithms.

CO10. Synthesize new graph algorithms and algorithms that employ graph computations as key components, and analyze them.

Course Outcomes of the Course "Embedded Systems"

CO1. Classify system and embedded system and describe various types of processors.

CO2. Describe design process of an Embedded System.

CO3. Describe different types of Communication Protocols.

CO4. Classify interrupts and Interrupt Service Routine.

CO5. Describe Context Switching, Latency and Deadline.

CO6. Write down different programming modeling concepts.

CO7. Differentiate Threads, Tasks and ISRs.

CO8. Describe process management, file and I/O Subsystem Management.

CO9. Differentiate normal Operating System and Real Time Operating System.

CO10. Write down tools for developing Embedded System.

Course Outcomes of the Course "Java Technology"

CO1. Describe features of Object Oriented programming and Java.

CO2. Write down simple Java Programs.

CO3. Describe operators, expressions, iteration statements and jump statements.

CO4. Classify Class and Interface.

CO5. Describe the concept of inheritance, method Overloading and Method Overriding.
CO6. Describe Exception Handling in java.
CO7. Write down basics of Applet.
CO8. Describe Multi-Threaded Programming and Event Handling.
CO9. Classify Threading and Synchronization.
CO10. Describe how to access Database using Java.

Course Outcomes of the Course "Advanced Operating System"
CO1. Analyze the structure of OS and basic architectural components involved in OS design
CO2. Analyze and design the applications to run in parallel either using process or thread models of different OS
CO3. Understand and analyze theory and implementation of: processes, resource control, physical and virtual memory, scheduling, I/O and files.
CO4. Analyze the various device and resource management techniques for timesharing and distributed systems
CO5. Understand the Mutual exclusion, Deadlock detection and agreement protocols of Distributed operating   system
CO6. Interpret the mechanisms adopted for file sharing in distributed Applications
CO7. Conceptualize the components involved in designing a contemporary OS
CO8. Understand the high-level structure of the Linux kernel both in concept and source code
CO9. Identify use and evaluate the storage management policies with respect to different storage management technologies.
CO10. Evaluate the requirement for process synchronization and coordination handled by operating system

Course Outcomes of the Course "Internet of Things"
CO1. Explain the definition and usage of the term "Internet of Things" in different contexts
CO2. Understand the key components that make up an IoT system
CO3. Differentiate between the levels of the IoT stack and be familiar with the key technologies and
CO4. Protocols employed at each layer of the stack
CO5. Apply the knowledge and skills acquired during the course to build and test a complete, working IoT system involving prototyping, programming and data analysis
CO6. Explain where the IoT concept fits within the broader ICT industry and possible future trends appreciate the role of big data, cloud computing and data analytics in a typical   IoTsystem
CO7. Describe sensors, microcontrollers, Single Board Computers (SBCs) – Raspberry Pi as prototyping platform and communication interfaces to design and build   IoT devices.
CO8. Explain cloud infrastructure, services, APIs, and architectures of commercial and industrial   cloud platforms.
CO9. Describe Integration of IoT with enterprise and user applications (Mobile Phone /SCADA/Enterprise Apps)

Course Outcomes of the Course "Advanced Database Management System"
CO1. Classify object and object relational database.
CO2. Describe ODMG, ODL and OQL model.
CO3. Describe object relational features in SQL/oracle.
CO4. Distinguish between parallel and distributed databases.
CO5. Describe centralized and client server architecture.
CO6. Write down commit protocols.
CO7. Describe data mining and text mining.
CO8. Differentiate active databases, temporal databases deductive databases and spatial databases.
CO9. Describe advanced data models and data warehousing.
CO10. Describe XML databases and cloud based databases.

Course Outcomes of the Course "Mobile and Wireless Communications"
CO1. Explain the basic concepts of wireless network and wireless generations.
CO2. Demonstrate the different wireless technologies such as CDMA, GSM, GPRS etc
CO3. Describe and judge the emerging wireless technologies standards such as WLL,WLAN, WPAN,WMAN.
CO4. Explain the design considerations fordeploying thewireless network infrastructure.
CO5. Differentiate and support the security measures, standards. Services and layer wise security considerations.
CO6. Evaluate Ad-hoc networks and wireless sensor networks.
CO7. Identify the limitations of 2G and 2.5G wireless mobile communication and use design of 3G and beyond mobile communication systems
CO8. Describe various version of IEEE 802.11 Standards
CO9. Explain virtual Networks, Bluetooth Protocols
CO10. Explain IRIDIUM and GLOBALSTAR system and its features.

Course Outcomes of the Course "Linux Environmental System"
CO1. Customize a LINUX login account using environment variables, configuration files and startup scripts
CO2. Maintain LINUX directories and files, manage LINUX jobs and processes, use of LINUX pipes and file redirection,
CO3. Manipulate data with proper use of LINUX filters, role of an operating system and LINUX philosophy
CO4. Operate in both graphical and text-based environments; automate a sequence of operations by writing a shell script.
CO5. Apply LINUX security tools to ensure LINUX directories and files are protected from unauthorized users.
CO6. Relate the use of on-line documentation, research and experimentation in order to discover how new LINUX commands function
CO7. Describe LINUX System Administration and Network Administration
CO8. Demonstrate how to utilize Text Editors
CO9. Identify shell features and environmental customization and create shell scripts
CO10. Explain C / C++ programs on LINUX.

Course Outcomes of the Course ".NET Technology"
CO1. Write down previous state of affairs and building blocks of .NET.
CO2. Describe role of Common Intermediate Language and Common Language Runtime.
CO3. Describe base class libraries and namespaces.
CO4. Write down simple c# application.
CO5. Classify command line compiler and command line debugger.
CO6. Describe C# preprocessor directives and response files.
CO7. Classify objects, constructors, static methods.
CO8. Describe object oriented programming concepts using c#.
CO9. Describe exceptions and exception handling methodologies.

Course Outcomes of the Course "Computer Graphics and Multimedia"
CO1. Describe the fundamentals of the modern GPU programming pipeline
CO2. Explain the Essential Mathematics in Computer Graphics
CO3. Describe Color and light representation and manipulation in graphics systems
CO4. Explain Common data structures to represent and manipulate geometry
CO5. Describe Common approaches to model light and materials

CO6. Describe Basic Image-Processing Techniques
CO7. Describe Basic Shading Techniques
CO8. Explain how the human visual system plays a role in interpretation of graph
CO9. Describe applications and technologies for Multimedia
CO10. Create a movie using Simple Animation

Course Outcomes of the Course "Software Engineering"
CO1. Understanding professional, ethical and social responsibility of a software
CO2. Classify software applications and Identify unique features of various domains
CO3. Plan, schedule and execute a project considering the risk management.
CO4. Classify software applications and Identify unique features of various domains.
CO5. Explain needs for software specifications and different types of software
CO6. Design test cases of a software system.
CO7. Distinguish among SCM and SQA and can classify different testing strategies
CO8. Apply the software testing techniques in commercial environment
CO9. Justify role of SDLC in Software Project Development
CO10. Implement various test processes for quality improvement.

Course Outcomes of the Course "Cloud Computing"
CO1. Understanding the key dimensions of the challenge of Cloud Computing
CO2. Define Cloud Computing and memorize the different Cloud Service and Deployment Models
CO3. Describe importance of virtualization along with their technologies.
CO4. Use and Examine different cloud computing services.
CO5. Analyze the components of open stack & Google Cloud platform and understand Mobile Cloud Computing.
CO6. Describe the key components of Amazon web Service.
CO7. Design & develop backup strategies for cloud data based on features.
CO8. Create combinatorial auctions for cloud resources and design scheduling algorithms for computing clouds.
CO9. Apply and design suitable Virtualization concept, Cloud Resource Management and design scheduling algorithms
CO10. Design different workflows according to requirements and apply map reduce programming model.
CO11. Analyze case studies to derive the best practice model to apply when developing and deploying cloud based applications.
CO12. Create a house like figure and rotate it about a given fixed point using OpenGL functions in OpenGL.
CO12. Implement the Cohen-Sutherland line clipping algorithm. to specify the input line, window for clipping and viewport for displaying the clipped image in OpenGL

Course Outcomes of the Course "Mobile E-Commerce"
CO1. Describe traditional commerce and E-commerce – Internet and WWW
CO2. Explain Role of E-commerce, Packet Switched Networks- TCP/IP protocol Script
CO3. Explain Infrastructure of M–Commerce, Types of Mobile Commerce Services – Technologies of Wireless Business
CO4. Explain Wireless Application Protocol, Mobile Business Services in Mobile E-commerce
CO5. Describe Mobile Data Technologies, Adoption and Diffusion of Small Business, E–commerce in The Automotive Industry
CO6. Write down the role of Mobile Advertising in Building a Brand M– commerce Business Models.
CO7. Classify Wireless Personal And Local Area Networks
CO8. Explain Strategy Formulation In Mobile Communications Networks.
CO9. Describe Asset Tracking and Maintenance/Management in Business
CO10. Explain business to business mobile e– commerce

Course Outcomes of the Course "Project Work"

CO1. Discover potential research areas in the field of IT.
CO2. Conduct a survey of several available literature in the preferred field of study
CO3. Compare and contrast the several existing solutions for research challenge
CO4. Demonstrate an ability to work in teams and manage the conduct of the research study.
CO5. Formulate and propose a plan for creating a solution for the research plan identified.
CO6. To report and present the findings of the study conducted in the preferred domain.
CO7. Design & develop the software projects
CO8. Identify risks; manage the change to assure quality in software projects.
CO9. Apply testing principles on software project and understand the maintenance concepts.
CO10. Design and communicate ideas about software system solutions at different levels