SRI VENKATESHWARA
COLLEGE OF ENGINEERING

Department of Computer Science and Engineering

COURSE OUTCOME

2022 Scheme

Third Semester

1. MATHEMATICS FOR COMPUTER SCIENCE (BCS301)

Course Outcomes:

CO1. Explain the basic concepts of probability, random variables, probability distribution
CO2. Apply suitable probability distribution models for the given scenario
CO3. Apply the notion of a discrete-time Markov chain and n-step transition probabilities to solve the given problem
CO4. Use statistical methodology and tools in the engineering problem-solving process
CO5. Compute the confidence intervals for the mean of the population
CO6. Apply the ANOVA test related to engineering problems

2. DIGITAL DESIGN AND COMPUTER ORGANIZATION (BCS302)

Course Outcomes:

CO1: Apply the K–Map techniques to simplify various Boolean expressions.
CO2: Design different types of combinational and sequential circuits along with Verilog programs.
CO3: Describe the fundamentals of machine instructions, addressing modes and Processor performance.
CO4: Explain the approaches involved in achieving communication between processor and I/O devices.
CO5: Analyze internal Organization of Memory and Impact of cache/Pipelining on Processor Performance.

3. OPERATING SYSTEMS (BCS303)

Course Outcomes:

CO1. Explain the structure and functionality of operating system

CO2. Apply appropriate CPU scheduling algorithms for the given problem.
CO3. Analyse the various techniques for process synchronization and deadlock handling.
CO4. Apply the various techniques for memory management
CO5. Explain file and secondary storage management strategies.
CO6. Describe the need for information protection mechanisms

4. DATA STRUCTURES AND APPLICATIONS (BCS304)

Course Outcomes
CO1. Explain different data structures and their applications.
CO2. Apply Arrays, Stacks and Queue data structures to solve the given problems.
CO3. Use the concept of linked list in problem solving.
CO4. Develop solutions using trees and graphs to model the real-world problem.
CO5. Explain the advanced Data Structures concepts such as Hashing Techniques and Optimal Binary Search Trees.

5. OBJECT-ORIENTED PROGRAMMING WITH JAVA (BCS306A)

Course Outcomes
CO1.Demonstrate proficiency in writing simple programs involving branching and looping structures.
CO2.Design a class involving data members and methods for the given scenario.
CO3.Apply the concepts of inheritance and interfaces in solving real world problems.
CO4.Use the concept of packages and exception handling in solving complex problem
CO5.Apply concepts of multithreading, autoboxing and enumerations in program development

6. OBJECT-ORIENTED PROGRAMMING WITH C++ (BCS306B) (ALTERNATE TO BCS306A)

Course Outcomes
CO1.Illustrate the basic concepts of object-oriented programming. CO2. Design appropriate classes for the given real world scenario. CO3. Apply the knowledge of compile-time / run-time polymorphism to solve the given problem CO4 Use the knowledge of inheritance for developing optimized solutions CO5 Apply the concepts of templates and exception handling for the given problem CO6 Use the concepts of input output streams for file operations

7. SOCIAL CONNECT AND RESPONSIBILITY (BSCK307)

Course Outcomes
CO1: Communicate and connect to the surrounding. CO2: Create a responsible connection with the society. CO3: Involve in the community in general in which they work. CO4: Notice the needs and problems of the community and involve them in problem –solving. CO5: Develop among themselves a sense of social & civic responsibility & utilize their knowledge in finding practical solutions to individual and community problems. CO6: Develop competence required for group-living and sharing of responsibilities & gain skills in mobilizing community participation to acquire leadership qualities and democratic attitudes.

Fourth Semester

1. ANALYSIS & DESIGN OF ALGORITHMS (BCS401)

Course Outcomes:

CO1. Apply asymptotic notational method to analyze the performance of the algorithms in terms of time complexity.
CO2. Demonstrate divide & conquer approaches and decrease & conquer approaches to solve computational problems.
CO3. Make use of transform & conquer and dynamic programming design approaches to solve the given real world or complex computational problems.
CO4. Apply greedy and input enhancement methods to solve graph & string based computational problems.
CO5. Analyse various classes (P,NP and NP Complete) of problems
CO6. Illustrate backtracking, branch & bound and approximation methods.

2. MICROCONTROLLERS (BCS402)

Course Outcomes:

CO1: Explain the ARM Architectural features and Instructions.
CO2: Develop programs using ARM instruction set for an ARM Microcontroller.
CO3: Explain C-Compiler Optimizations and portability issues in ARM Microcontroller.
CO4: Apply the concepts of Exceptions and Interrupt handling mechanisms in developing applications.
CO5: Demonstrate the role of Cache management and Firmware in Microcontrollers.

3. DATABASE MANAGEMENT SYSTEM (BCS403)

Course Outcomes:

CO1: Describe the basic elements of a relational database management system
CO2: Design entity relationship for the given scenario.
CO3: Apply various Structured Query Language (SQL) statements for database manipulation.

CO4: Analyse various normalization forms for the given application.
CO5: Develop database applications for the given real world problem.
CO6: Understand the concepts related to NoSQL databases.

4. DISCRETE MATHEMATICAL STRUCTURES (BCS405A)

Course Outcomes
CO1: Apply concepts of logical reasoning and mathematical proof techniques in proving theorems and statements. CO2. Demonstrate the application of discrete structures in different fields of computer science. CO3. Apply the basic concepts of relations, functions and partially ordered sets for computer representations. CO4. Solve problems involving recurrence relations and generating functions. CO5. Illustrate the fundamental principles of Algebraic structures with the problems related to computer science & engineering.

5. OPTIMIZATION TECHNIQUE (BCS405C)

Course Outcomes
CO1: Apply the concepts of vector calculus to solve the given problem. CO2: Apply the concepts of partial differentiation in machine learning and deep neural networks. CO3: Analyze the convex optimization algorithms and their importance in computer science & engineering. CO4: Apply the optimization algorithms to solve the problem. CO5: Analyze the advanced optimization algorithms for machine learning .

6. GREEN IT AND SUSTAINABILITY (BCS456A)

Course Outcomes
CO1: Classify the challenges for Green ICT CO2: Relate the environmental impact due to emerging technologies.

CO3: Demonstrate different aspects of ICT metrics.
CO4: Compare the various parameters related to Sustainable Cloud Computing.
CO5: Interpret the effects of software design on the sustainability.

7. UI/UX (BCS456C)

Course Outcomes
CO1: Explain the user experience design requirements.
CO2: Relate design thinking concepts and mental models to UX design.
CO3: Illustrate UX design in line with design goals, metrics and targets.
CO4: Demonstrate different prototyping in relation with software engineering.
CO5: Explain UX design principles with case examples.

Fifth Semester

1. SOFTWARE ENGINEERING & PROJECT MANAGEMENT (BCS501):

Course Outcomes

CO1: Differentiate process models to judge which process model has to be adopted for the given scenarios.
CO2: Derive both functional and nonfunctional requirements from the case study.
CO3: Analyze the importance of various software testing methods and agile methodology.
CO4 Illustrate the role of project planning and quality management
CO5: Identify appropriate techniques to enhance software quality.

2. COMPUTER NETWORKS (BCS502):

Course Outcomes

CO1: Explain the fundamentals of computer networks.
CO2: Apply the concepts of computer networks to demonstrate the working of various layers and protocols in communication network.
CO3: Analyze the principles of protocol layering in modern communication systems.
CO4: Demonstrate various Routing protocols and their services using tools such as Cisco packet tracer

3. THEORY OF COMPUTATION (BCS503):

Course Outcomes

CO1: Apply the fundamentals of automata theory to write DFA, NFA, Epsilon-NFA and conversion between them.
CO2: Prove the properties of regular languages using regular expressions.
CO3: Design context-free grammars (CFGs) and pushdown automata (PDAs) for formal languages.

CO4: Design Turing machines to solve the computational problems.
CO5: Explain the concepts of decidability and undecidability.

4. COMPUTER GRAPHICS (BAI515A):

Course Outcomes
CO1: Explain the fundamentals of computer graphics systems. CO2: Develop event driven graphical applications by interfacing hardware devices. CO3: Apply the Geometrical Transformations on geometrical objects. CO4: Apply the concepts of viewing, lighting and shading on graphical objects. CO5: Demonstrate algorithms for 2D graphical primitives.

5. ARTIFICIAL INTELLIGENCE (BCS515B):

Course Outcomes
CO1: Explain the architecture and components of intelligent agents, including their interaction with the AI environment. CO2: Apply problem-solving agents and various search strategies to solve a given problem. CO3: Illustrate logical reasoning and knowledge representation using propositional and first-order logic. CO4: Demonstrate proficiency in representing knowledge and solving problems using first-order logic. CO5: Describe classical planning in the context of artificial intelligence, including its goals, constraints, and applications in problem-solving

6. UNIX SYSTEM PROGRAMMING (BCS515C):

Course Outcomes
CO1: Demonstrate the basics of Unix concepts and commands. CO2: Demonstrate the UNIX file system. CO3: Apply comands to reflect changes in file system.

CO4: Demonstrate IPC and process management.
CO5: Develop an application/service over a Unix system.

7. DISTRIBUTED SYSTEMS (BCS515D):

Course Outcomes
CO1: Identify the goals and challenges of distributed systems
CO2: Demonstrate the remote invocation techniques for communication
CO3: Describe the architecture of distributed file systems and name services
CO4: Apply clock synchronization algorithms to monitor and order the events.
CO5: Analyze the performance of mutual exclusion, election and consensus algorithms.
CO6: Illustrate the fundamental concepts and algorithms related to distributed transactions and replication

Sixth Semester

1. CLOUD COMPUTING (BCS601):

Course Outcomes

CO1: Describe various cloud computing platforms and service providers.
CO2: Illustrate the significance of various types of virtualization.
CO3: Identify the architecture, delivery models and industrial platforms for cloud computing based applications.
CO4: Analyze the role of security aspects in cloud computing.
CO5: Demonstrate cloud applications in various fields using suitable cloud platforms.

2. MACHINE LEARNING (BCS602):

Course Outcomes

CO1: Describe the machine learning techniques, their types and data analysis framework.
CO2: Apply mathematical concepts for feature engineering and perform dimensionality reduction to enhance model performance.
CO3: Develop similarity-based learning models and regression models for solving classification and prediction tasks.
CO4: Build probabilistic learning models and design neural network models using perceptrons and multilayer architectures
CO5: Utilize clustering algorithms to identify patterns in data and implement reinforcement learning techniques

3. Blockchain Technology (BCS613A):

Course Outcomes

CO1: Explain the Blockchain terminologies with its applications. design

CO2: Illustrate the working principles of Blockchain and the Smart Contract Lifecycle
CO3: Demonstrate the principles and methodologies used in Bitcoin
CO4: Develop Ethereum Network, Wallets, Nodes, Smart contract and DApps.
CO5: Make use of Hyperledger in Blockchain Based Application Architecture.

4. COMPUTER VISION (BCS613B):

Course Outcomes
CO1: Explain the fundamentals of computer vision and its applications.
CO2: Apply the image enhancement techniques for smoothing and sharpening of images.
CO3: Compare the different image restoration and segmentation techniques.
CO4: Demonstrate the smoothing and sharpening techniques for color images.
CO5: Explain morphological, feature extraction, and pattern classification techniques for object recognition.

5. COMPILER DESIGN (BCS613C):

Course Outcomes
CO1: Understand the different phases of compiler design techniques
CO2: Analyse the working of lexical analyser in design of compilers
CO3: Design syntax analyser using top down and bottom up approaches
CO4: Illustrate syntax-directed translation for a given grammar.
CO5: Explain intermediate code representation and code generation of compilers

6. ADVANCED JAVA (BCS613D):

Course Outcomes
CO1: Apply appropriate collection class/interface to solve the given problem
CO2: Demonstrate the concepts of String operations in Java
CO3: Apply the concepts of Swings to build Java applications

CO4: Develop web-based applications using Java servlets and JSP
CO5: Use JDBC to build database applications

7. INTRODUCTION TO DATA STRUCTURES (BCS654A):

Course Outcomes
CO1: Develop C programs utilizing fundamental concepts such as arrays, pointers and structures. CO2: Apply data structures like stacks and queues to solve problems. CO3: Develop C programs using linked lists and their various types. CO4: Explain the fundamental concepts of trees and their practical applications. CO5: Demonstrate different sorting and searching algorithms and determine their algorithmic complexities.

8. INTRODUCTION TO ARTIFICIAL INTELLIGENCE (BAI654D):

Course Outcomes
CO1: Identify the problems where the adaptation of AI has significant impact. CO2: Analyse the different approaches of Knowledge Representation. CO3: Explain Symbolic Reasoning under Uncertainty and Statistical reasoning. CO4: Derive the importance of different types of Learning Techniques. CO5: Explain Natural Language Processing and Expert System.

9. MOBILE APPLICATION DEVELOPMENT(BIS654C)

Course Outcomes
CO1: Explain Mobile Application Ecosystem like concepts, architecture, and lifecycle of mobile applications on Android CO2: Identify the key components of mobile application frameworks and development tools. CO3: Apply design principles to create intuitive and responsive user interfaces using appropriate UI/UX tools. CO4: Develop Functional Mobile Applications -Integrate core functionalities such as layouts, event handling, navigation, and multimedia support into applications. CO5: Implement local data storage mechanisms (SQLite, Shared Preferences) and external databases (Firebase, APIs) for mobile applications.