Department of Computer Science

Course: Bachelor of Computer Science (BSc-CS)

PROGRAM OUTCOME (POs)

A Bachelor of Science in Computer Science (BSc CS) program typically prepares students for careers in the field of computer science. The program covers a wide range of topics related to the design, development, and use of computer systems and software. Outcomes of a BSc CS program include:

1. **Technical skills:** Students should develop strong technical skills in areas such as programming languages, algorithms and data structures, computer networks, operating systems, and databases.

2. **Problem-solving skills:** Students should be able to analyze complex problems and develop solutions using computer science tools and techniques.

3. **Communication skills:** Students should be able to communicate effectively with colleagues and stakeholders, both verbally and in writing.

4. **Teamwork and collaboration**: Students should learn to work effectively in teams and collaborate with others to develop and implement computer science solutions.

5. **Ethical and professional conduct:** Students should be aware of ethical issues in computer science and demonstrate professional conduct in their work.

6. **Lifelong learning:** Students should have a strong foundation in computer science that will enable them to adapt to new technologies and continue learning throughout their careers Overall, a BSc CS program aims to prepare students to be competent, skilled, and ethical professionals in the field of computer science. Graduates of the program should be able to apply their knowledge and skills to solve real-world problems and advance the field of computer science.

Department of Computer Science

Course: Bachelor of Computer Science (BSc-CS)

Program Specific Outcome (POSs)

- Graduates will have a strong foundation in programming languages and be able to design, implement, and troubleshoot software applications.
- Algorithms and Data Structures: Graduates should have a solid understanding of algorithms and data structures, enabling them to solve complex problems efficiently.
- Computer Systems: Understanding the architecture and components of computer systems, including hardware and operating systems.
- Database Management: Graduates should be proficient in database design and management, including the ability to create and maintain databases for various applications.
- Software Development Practices: Knowledge of software development methodologies, version control systems, and best practices in software engineering.
- Networking: Understanding of computer networks, protocols, and network security, including the design and maintenance of networked systems.
- Web Development: Depending on the program, graduates may have acquired skills in web development, including front-end and back-end technologies.
- Artificial Intelligence and Machine Learning: Basic understanding of AI and machine learning concepts and applications.
- Cybersecurity: Awareness of cybersecurity principles, including securing systems and applications from potential threats.
- Operating Systems: Proficiency in understanding and working with various operating systems, including both desktop and server environments.
- Human-Computer Interaction (HCI): Understanding of HCI principles to design user-friendly and effective interfaces.

Department of Computer Science

Course: Bachelor of Computer Science (BSc-CS)

COURSE OUTCOME (COs)

FYB Sc- CS SEMESTER - I

USCS101 Computer Organization and Design

1. Understand the basic components and functioning of computer systems, including input/output devices, memory, processor, and storage devices.
2. Develop an understanding of digital logic design and its application in building circuits, including combinational and sequential logic circuits.
3. Understand the architecture and organization of a computer system, including the role of the processor, memory, and input/output devices.
4. Develop an understanding of assembly language programming, including instruction formats, addressing modes, and instruction set architecture.
5. Understand the basics of operating system concepts, including process management, memory management, and file system.

6. Analyze the performance of computer systems, including the role of cache memory, virtual memory, and disk scheduling algorithms.

USCS102 Programming with Python- I
1. Students will be able to understand the basic concepts of programming such as variables, data types, control structures, functions, and objects.
2. Students will be able to use Python libraries to solve problems related to data analysis, data visualization, scientific computing, and machine learning.
3. Students will learn how to use Python to manipulate data, including reading and writing data to files, processing data in memory, and cleaning and transforming data.
4. Students will understand the principles of object-oriented programming and be able to write Python programs using classes, objects, and inheritance.
5. Students will learn how to use Python to build web applications, including using web frameworks such as Flask or Django.
6. Students will be able to use Python to work with databases, including connecting to databases, querying data, and modifying data.
7. Students will understand how to use Python for network programming, including socket programming, client-server communication, and web scraping.

USCS103 Free and Open Source Software

1) Learner will understand the difference between open-source software and commercialsoftware.
2) Understand the policies, licensing procedures and ethics of FOSS.
3) Understand open-source philosophy, methodology and ecosystem.
4) Awareness with Open-Source Technologies.
5) Differentiate between Open Source and Proprietary software and Licensing.
6) Recognize the applications, benefits and features of Open-Source Technologies Gainknowledge to start, manage open-source projects.

USCS104 Database Systems

Upon completion of this course, students should be able to:

1. Understand the fundamental concepts of database management systems, includingdata modeling, database design, and relational databases.
2. Design and implement database schemas and tables using SQL, and perform various SQL operations such as selecting, inserting, updating, and deleting data.
3. Understand and apply the principles of normalization to ensure data integrity andeliminate data redundancy in database design.
4. Develop an understanding of data storage and retrieval techniques, including indexing,sorting, and hashing.
5. Understand the principles of database security, including access control, authentication, and authorization, and implement them in a database environment.
6. Analyze and evaluate the performance of database queries and transactions, and optimizethem to improve system performance.

USCS105 Discrete Mathematics

1. To understand the concepts of mathematical logic for analyzing propositions via truth tablesand proving theorems.
2. Apply sets for analyze & solving applied problems, and apply the properties of setoperations algebraically.
3. Help to understand the purpose of mathematical induction is to be able to prove an equationworks for all natural numbers and help how to find the recursive formula of a sequence
4. Understand the relations on sets and their properties.
5. Investigate functions as relations and their properties and Evaluate functions as relations andtheir properties.
6. To understand the basic concepts of graphs, digraphs, and trees.

USCS106 Descriptive Statistics and Introduction to Probability

1. To understand learning ability to evaluate different types of data.
2. To understand measuring of Central Tendency and dispersion. To present a brief picture of data.
3. To evaluate various aspects of a distribution, such as its central tendency, skewness, and kurtosis.
4. To understand how to find correlation of different variable and least squares regression. To apply regression to find extra relation between them.
5. To analyze statistical data using R software.

USCS107 Soft Skills Development

1) Learner will be able to understand the significance and essence of a wide range of soft skills.
2) Learn how to apply soft skills in a wide range of routine social and professional settings.
3) Learn how to employ soft skills to improve interpersonal relationships.
4) Learn how to employ soft skills to enhance employability and ensure workplace and career success.
5) Learners will develop skills for Academic and Professional Presentations.
6) Learners will able to understand Leadership Qualities and Ethics.
7) Ability to understand the importance of stress management in their academic &professional life.

FYCS Semester II

USCS201 Programming With C

1. Apply the fundamental concepts of programming to write simple programs in C.
2. Understand the history and evolution of programming languages and models.
3. Synthesize programming logic using pseudocode and flowchart symbols to develop programs.
4. Evaluate program logic to identify and fix errors and bugs.
5. Create efficient and effective programs by applying programming concepts and techniques.
6. Understand the program development cycle and the importance of debugging and testing.
7. Apply conditional statements and loops to decision making within Analyze the performance of computer systems, including the role of cache memory, virtual memory, and disk scheduling algorithms.

USCS102 Programming with Python- I

1. Students will be able to understand the basic concepts of programming such as variables, data types, control structures, functions, and objects.
2. Students will be able to use Python libraries to solve problems related to data analysis, data visualization, scientific computing, and machine learning.
3. Students will learn how to use Python to manipulate data, including reading and writing data to files, processing data in memory, and cleaning and transforming data.
4. Students will understand the principles of object-oriented programming and be able to write Python programs using classes, objects, and inheritance.

5. Students will learn how to use Python to build web applications, including using web frameworks such as Flask or Django.
6. Students will be able to use Python to work with databases, including connecting to databases, querying data, and modifying data.
7. Students will understand how to use Python for network programming, including socketprogramming, client-server communication, and web scraping.

USCS103 Free and Open Source Software
1. Learner will understand the difference between open-source software and commercialsoftware.
2. Understand the policies, licensing procedures and ethics of FOSS.
3. Understand open-source philosophy, methodology and ecosystem.
4. Awareness with Open-Source Technologies.
5. Differentiate between Open Source and Proprietary software and Licensing.
6. Recognize the applications, benefits and features of Open-Source Technologies Gainknowledge to start, manage open-source projects.

USCS104 Database Systems
Upon completion of this course, students should be able to:
1. Understand the fundamental concepts of database management systems, includingdata modeling, database design, and relational databases.
2. Design and implement database schemas and tables using SQL, and perform various SQL operations such as selecting, inserting, updating, and deleting data.
3. Understand and apply the principles of normalization to ensure data integrity andeliminate data redundancy in database design.
4. Develop an understanding of data storage and retrieval techniques, including indexing,sorting, and hashing.
5. Understand the principles of database security, including access control, authentication, and authorization, and implement them in a database environment.
6. Analyze and evaluate the performance of database queries and transactions, and optimizethem to improve system performance.

USCS105 Discrete Mathematics
1. To understand the concepts of mathematical logic for analyzing propositions via truth tablesand proving theorems.
2. Apply sets for analyze & solving applied problems, and apply the properties of setoperations algebraically.
3. Help to understand the purpose of mathematical induction is to be able to prove an equationworks for all natural numbers and help how to find the recursive formula of a sequence
4. Understand the relations on sets and their properties.
5. Investigate functions as relations and their properties and Evaluate functions as relations andtheir properties.
6. To understand the basic concepts of graphs, digraphs, and trees.

USCS106 Descriptive Statistics and Introduction to Probability

1. To understand learning ability to evaluate different types of data.
2. To understand measuring of Central Tendency and dispersion. To present a brief picture of data.
3. To evaluate various aspects of a distribution, such as its central tendency, skewness, and kurtosis.
4. To understand how to find correlation of different variable and least squares regression. To apply regression to find extra relation between them.
5. To analyze statistical data using R software.

USCS107 Soft Skills Development

8) Learner will be able to understand the significance and essence of a wide range of soft skills.
9) Learn how to apply soft skills in a wide range of routine social and professional settings.
10) Learn how to employ soft skills to improve interpersonal relationships.
11) Learn how to employ soft skills to enhance employability and ensure workplace and career success.
12) Learners will develop skills for Academic and Professional Presentations.
13) Learners will able to understand Leadership Qualities and Ethics.
14) Ability to understand the importance of stress management in their academic &professional life.

FYCS Semester II

USCS201 Programming With C

1. Apply the fundamental concepts of programming to write simple programs in C.
2. Understand the history and evolution of programming languages and models.
3. Synthesize programming logic using pseudocode and flowchart symbols to develop programs.
4. Evaluate program logic to identify and fix errors and bugs.
5. Create efficient and effective programs by applying programming concepts and techniques.
6. Understand the program development cycle and the importance of debugging and testing.
7. Apply conditional statements and loops to decision making within a program.
8. Understand the use and implementation of arrays and strings in programming.
9. Evaluate program structure and design to ensure readability, maintainability and scalability.
10. Create and use functions and pointers to modularize and optimize program code.

USCS202 Programming With Python-II

1. Students will be able to understand the basic concepts of programming such as variables, data types, control structures, functions, and objects.
2. Students will be able to use Python libraries to solve problems related to data analysis, data visualization, scientific computing, and machine learning.
3. Students will learn how to use Python to manipulate data, including reading and writing data to files, processing data in memory, and cleaning and transforming data.
4. Students will understand the principles of object-oriented programming and be able to write Python programs using classes, objects, and inheritance.
5. Students will learn how to use Python to build web applications, including using web frameworks such as Flask or Django.
6. Students will be able to use Python to work with databases, including connecting to databases,

querying data, and modifying data.

7. Students will understand how to use Python for network programming, including socket programming, client-server communication, and web scraping.

USCS203 Linux

1. Students will understand the fundamental concepts of operating systems and their components, including the Linux kernel, file system, and shell.
2. Students will be able to use the Linux command line interface to navigate the file system, manipulate files and directories, and execute commands.
3. Students will be able to configure and manage user accounts and permissions, including setting user and group permissions, and managing file ownership.
4. Students will learn how to install, configure, and manage Linux packages and applications, including using package managers and software repositories.
5. Students will be able to configure and manage Linux system services, including network configuration, system logs, and process management.

USCS204 Data Structures

1. Understanding the fundamental concepts of data structures: Students will learn the basic concepts of data structures such as arrays, linked lists, stacks, queues, trees, graphs, and their properties.
2. Ability to analyze algorithms: Students will be able to analyze the time and space complexity of various algorithms and select appropriate data structures to optimize their performance.
3. Ability to implement data structures: Students will be able to implement various data structures using programming languages such as C, C++, Java, or Python.
4. Ability to solve problems using data structures: Students will be able to use appropriate data structures to solve various problems such as searching, sorting, and graph algorithm.

5. Ability to design and implement complex data structures: Students will be able to design and implement complex data structures such as hash tables, B-trees, and priority queues.

USCS205 Calculus

1. The objective is to introduced basic tools of calculus. Which is used to evaluate real world problems.
2. Students should understand the relationship between the derivative and the definite integral.
3. Students should understand the meaning of the derivative terms of a rate of change and local linear approximation and should be able to use derivatives to solve a variety of problems.
4. To understand definition of functions in partial derivative in logical way.
5. To understand the meaning of the derivative in terms of a rate of change and local linear approximation and should be able to use derivatives to solve a variety of problems.

USCS206 Statistical Methods and Testing Hypothesis

1) Learner will be aware about basic probability axioms and rules and its application.
2) They will be able to understand the concept of conditional probability and Independence of events.
3) Learner will be familiar with discrete and continuous random variables as well as standard discrete and continuous distributions.
4) They will learn computational skills to implement various statistical inferential approaches.
5) After successful completion of this course, learners would be able to Calculate probability, conditional probability and independence.

6) Apply the given discrete and continuous distributions whenever necessary.
7) Define null hypothesis, alternative hypothesis, level of significance, test statistic and p value.
8) Perform Test of Hypothesis as well as calculate confidence interval for a population parameter for single sample and two sample cases.
9) Apply non-parametric test whenever necessary. Conduct and interpret one- way and two-way ANOVA.

USCS207 Green Technologies

1. Define Green IT and its significance in the modern era.
2. Understand the principles of sustainable computing and the role of IT in achieving sustainability goals.
3. Describe the energy consumption of computing systems and evaluate the environmental impact of computing.
4. Analyze the benefits and drawbacks of virtualization, cloud computing, and green datacenters.
5. Identify and assess various strategies for reducing the environmental impact of computing systems, such as power management, recycling, and green procurement.
6. Design and implement eco-friendly computing systems, and evaluate their effectiveness in reducing energy consumption and environmental impact.
7. Apply the principles of green IT in the development of soft aware, including the use of eco-design and sustainable programming practices.

SY B Sc CS SEMESTER-III

USCS301 Theory of Computation

1) Students will understand grammar, languages and other elements of modern language design.
2) Also, they will able to develop capabilities to design and develop formulations for computing models and identify its applications in diverse areas.
3) They will understand Grammar and Languages about Automata theory and its application in Language Design.
4) Learn about Turing Machines and Pushdown Automata.
5) Understand Linear Bound Automata and its applications.

USCS302 Core Java

1. Understand the features of Java programming language and apply them to write basic programs.
2. Apply object-oriented programming concepts like inheritance, polymorphism, and encapsulation to design and develop robust software systems.
3. Evaluate the importance of exception handling in Java and create custom exceptions to handle errors.
4. Synthesize the concepts of multithreading, synchronization, and wait/notify methods to develop concurrent applications.

5. Create I/O streams to read and write data from/to files and serialize objects for storage and retrieval.
6. Understand the basics of networking in Java and create client-server communication programs using sockets.
7. Apply the concept of wrapper classes to convert primitive data types into objects and vice versa.
8. Evaluate the Collection Framework and its interfaces like List, Set, and Map to develop efficient data structures.
9. Synthesize the concept of inner classes and apply them to write nested and anonymous classes.
8. Create user interfaces using AWT components, layouts, and event handling mechanisms.

USCS303 Operating System

1. Understand the fundamental concepts of operating systems, including process management, memory management, file systems, and input/output operations.
2. Understand the different types of operating systems, including batch, time-sharing, and real-time operating systems, and their applications in various computing environments.
3. Understand the principles of process management, including process scheduling, synchronization, and communication, and develop skills in process management using operating system tools.
4. Develop skills in memory management, including virtual memory, paging, and segmentation, and understand the impact of memory management on system performance.
5. Understand the principles of file systems and disk management, including file organization, file access methods, and disk scheduling, and develop skills in file system management using operating system tools.
6. Understand the principles of input/output management, including device drivers, interrupts, and buffering, and develop skills in input/output management using operating system tools.

USCS304 Database Management systems

1. Understand the fundamental concepts of database management systems, including data modeling, database design, and relational databases.
2. Design and implement database schemas and tables using SQL, and perform various SQL operations such as selecting, inserting, updating, and deleting data.
3. Understand and apply the principles of normalization to ensure data integrity and eliminate data redundancy in database design.
4. Develop an understanding of data storage and retrieval techniques, including indexing, sorting, and hashing.
5. Understand the principles of database security, including access control, authentication, and authorization, and implement them in a database environment.
6. Analyze and evaluate the performance of database queries and transactions, and optimize them to improve system performance.

USCS305 Combinatorics and Graph Theory

1. The basic principles of combinatorics, including counting principles, permutations, combinations, and generating functions.
2. The basic principles of graph theory, including graph representation, connectivity, coloring, planarity, and graph algorithms.
3. How to apply combinatorial and graph theory techniques to solve problems in various fields, such as computer science, engineering, and science.
4. How to design and analyze algorithms that use combinatorial and graph theory techniques.
5. How to use computational tools to solve problems in combinatorics and graph theory.
6. How to write rigorous proofs of combinatorial and graph theory results.
7. How to communicate mathematical ideas and results effectively, both orally and in writing.

USCS306 Physical Computing and IoT Programming

1. Define physical computing and IoT (Internet of Things) and explain their significance in modern society.
2. Identify the various components used in physical computing, including microcontrollers, sensors, actuators, and communication protocols.
3. Design and implement simple circuits using breadboards, resistors, capacitors, LEDs, and other electronic components.
4. Develop basic programming skills in languages such as Python or C++ to control and interact with physical computing devices.
5. Build and program a range of physical computing devices, including robots, interactive installations, and smart objects.
6. Understand the principles of IoT and its various applications in different industries, including smart homes, healthcare, and transportation.
7. Analyze the security and privacy risks associated with IoT devices and develop strategies to mitigate these risks.
8. Develop skills in using cloud platforms, APIs, and web services to collect and process data from IoT devices.
9. Design and implement IoT projects using wireless communication technologies such as Bluetooth, Wi-Fi, and LoRa WAN.

USCS307 Skill Enhancement: Web Programming

1. Students will understand the principles of web programming, including web development frameworks, HTML, CSS, and JavaScript.
2. Students will be able to create and design web pages using HTML and CSS, including understanding web layout, typography, and responsive design.
3. Students will be able to write JavaScript code to enhance the functionality of web pages, including interactivity, user input validation, and dynamic content creation.
4. Students will learn how to use web development frameworks such as React, Angular, or Vue.js to create scalable and maintainable web applications.
5. Students will understand how to use server-side technologies such as PHP, Node.js, or ASP.NET to develop dynamic web applications.
6. Students will be able to integrate web applications with databases, including understanding database design, SQL, and database administration.

SYCS Semester IV

USCS401 Fundamentals of Algorithms

1) Understand the basic concepts and terminology related to algorithms, including complexity analysis and data structures.
2) Analyze and design algorithms for a variety of problems, including searching, sorting, and graph algorithms.
3) Develop an understanding of the different types of data structures, including arrays, linked lists, trees, and graphs.
4) Apply algorithmic design techniques, such as dynamic programming, greedy algorithms, and divide-and-conquer, to solve problems.
5) Understand the importance of algorithmic efficiency and be able to evaluate the efficiency of an algorithm using big-O notation.
6) Develop an understanding of common algorithm design patterns, such as recursion, backtracking, and branch-and-bound.

USCS402 Advanced Java

1. Understand the Swing components and their hierarchy, and distinguish between AWT and Swing components.
2. Apply JDBC to connect Java applications to databases, and utilize various types of resultsets, including Read Only, Updatable, Forward Only, and Scrollable.
3. Evaluate different modes of JDBC connections, including Prepared statement, BatchUpdates, and Callable Statement, and use BLOB and CLOB data types.
4. Understand the Servlets and JSP life cycle, and their interaction with web servers and containers.
5. Apply Session Tracking Mechanisms in Servlets to maintain user state across multiple requests.
6. Evaluate the difference between JSP scripting elements, directives, standard actions, and customized actions.
7. Understand the JavaBeans Properties and their usage, and create JavaBeans in Java.
8. Synthesize the MVC Architecture and apply the Struts 2 framework features, including the MVC pattern, Actions, Interceptors, Results & Result Types, and Value Stack/OGNL.
9. Create JSON documents using Java, and compare JSON with XML.

USCS403 Computer Networks

1. An understanding of the basics of computer networking, including the OSI and TCP/IP models.
2. Knowledge of different network topologies and architectures, including LAN, WAN, and MAN.
3. Familiarity with network protocols such as Ethernet, TCP/IP, HTTP, SMTP, and FTP.
4. The ability to design and configure networks, including subnetting and addressing schemes.
5. An understanding of network security concepts, including authentication, authorization, and encryption.
6. An understanding of wireless networking technologies such as Wi-Fi, Bluetooth, and cellular networks.

7. Knowledge of network management and monitoring tools.
8. An understanding of emerging networking technologies such as Software-DefinedNetworking (SDN) and Network Functions Virtualization (NFV).
9. The ability to troubleshoot network issues using various diagnostic tools.
10. The ability to work collaboratively in a team to design, implement, and maintain a network infrastructure.

USCS404 Software Engineering
1. Understand the fundamental concepts of software engineering, including softwaredevelopment processes, software requirements, and software design.
2. Develop an understanding of software development methodologies, including Waterfall, Agile, and DevOps, and understand the advantages and disadvantages of each.
3. Understand the principles of software project management, including projectplanning, estimation, and tracking, and develop skills in managing software development projects.
4. Understand and apply software requirements engineering techniques, including requirement elicitation, analysis, and specification, to develop software that meetscustomer needs.

USCS405 Linear Algebra Using Python
1. To learn the concept of complex number in python programming.
2. To understand and solve systems of linear equations using various methods includingGaussian and Gauss Jordan elimination and inverse matrices.
3. To understand relationship between coefficient matrix invariability and solutions to a systemof linear equations and the inverse matrices.
4. To understand real vector spaces and subspaces and apply their properties.
5. To find a basis for the row space, column space and null space of a matrix and find the rankand nullity of a matrix.
6. To understand how to find eigenvalues and eigenvectors and use them in applications.

USCS406 .Net Technologies
1) Learner will be able to explore .NET technologies for designing and developing dynamic, interactive and responsive web applications.
2) They will Understand the .NET framework and core concept and components of the .net framework, including its architecture, CLR(Common Language Runtime) and BCL.
3) Learner will develop a proficiency in the C# programming language.
4) Proficiently develop ASP.NET web applications using C#.
5) Students will learn how to use .Net technologies to develop applications, includingcreating user interfaces, accessing databases and writing code using C# and other programming languages.
6) Use ADO.NET for data persistence in a web application.
7) Familiarity with IDE (Integrated Development Environment) for developing .Netapplications.

USCS407 Skill Enhancement: Android Developer Fundamentals

1. Understanding of Android Development: Students should be able to understand the basic concepts, principles, and practices of Android app development, including the Android application architecture, activities, layouts, resources, and intents.
2. Ability to develop Android applications: Students should be able to develop Android applications using Java and Android Studio, including the ability to design and implement effective user interfaces, data models, and business logic.
3. Familiarity with Android Components: Students should be familiar with Android components such as Broadcast Receivers, Services, and Content Providers, and understand how to use them in app development.
4. Knowledge of Android libraries and frameworks: Students should be familiar with popular Android libraries and frameworks such as Retrofit, OkHttp, and RxJava, and understand how to use them to build efficient and effective Android applications.
5. Understanding of Android Material Design: Students should be able to understand the principles of Android Material Design and apply them to develop visually appealing and user-friendly Android applications.

TYCS Semester V

USCS502 Linux Server Administration

1. Students will be able to install, configure and manage a Linux server operating system, including user and group management, file systems, and security settings.
2. Students will understand the principles of networking and be able to configure networking settings on a Linux server, including IP addresses, network interfaces, and routing.
3. Students will be able to install, configure and manage common Linux server software, such as web servers (Apache, Nginx), database servers (MySQL, PostgreSQL), and email servers (Postfix, Sendmail).
4. Students will learn how to use command-line tools to manage and troubleshoot a Linux server, including utilities such as grep, awk, sed, and find.
5. Students will learn how to write basic shell scripts to automate common server administration tasks.

USCS503 Software Testing and Quality Assurance

1. Upon completion of this course, students should be able to:
2. Understand the fundamental concepts of software testing, including software testing principles, software testing life cycle, and types of software testing.
3. Design and implement test cases and test plans, and perform various types of testing such as unit testing, integration testing, system testing, and acceptance testing.
4. Understand and apply different software testing techniques, such as white-box testing, black-box testing, and gray-box testing.
5. Analyze and evaluate software metrics, such as defect density, code coverage, and code complexity, and use them to assess the quality of software.
6. Understand the principles of quality control, including continuous integration, version control, and bug tracking, and apply them in a software development environment.

7.   Understand the role of software quality assurance in software development, including quality standards, quality control processes, and quality improvement models.
8.   Develop an understanding of various software development models, such as Waterfall and Agile, and understand the role of software testing and quality assurance in each model.

USCS504 Information and Network Security

1.   Understand the fundamental concepts of cryptography, including encryption, decryption, symmetric key cryptography, and public key cryptography.
2.   Analyze and evaluate various cryptographic algorithms and protocols, such as AES, RSA, and SSL/TLS, and apply them to secure communication and data transfer.
3.   Understand the various types of attacks on cryptographic systems, including brute force attacks, dictionary attacks, and side-channel attacks, and apply appropriate countermeasures to protect against them.
4.   Understand the principles of network security, including the concepts of firewalls, intrusion detection/prevention systems, and virtual private networks (VPNs).
5.   Analyze and evaluate the security of network architectures and protocols, such as TCP/IP, DNS, and HTTP, and apply appropriate security measures to protect against common attacks.

USCS505 Web Services

1.   Understand the basics of web services and their types.
2.   Apply knowledge of distributed computing infrastructure, XML, and SOAP in building web services with JAX-WS.
3.   Develop and consume simple web services across platforms.
4.   Register and discover web services using Service Oriented Architecture.
5.   Evaluate the web services development life cycle and desirable program characteristics.
6.   Create JSON message formats and evaluate tools and frameworks around JSON.
7.   Develop and build RESTful web services with JAX-RS APIs.
8.   Analyze design guidelines for building secure RESTful web services.
9.   Synthesize core architectural elements of a RESTful system and discover RESTful web services.
10. Understand the fundamental concepts of WCF and its architecture.
11. Implement WCF programming and feature details.
12. Evaluate WCF and .NET Framework Client Profile for web services.
13. Develop and evaluate the quality of service for web services using WCF.

USCS506 Game Programming

1)   Understanding of game design principles: Students should be able to comprehend and apply fundamental game design principles, such as gameplay mechanics, level design, and user interface design.
2)   Knowledge of game engines and tools: Students should be familiar with different game engines, tools, and frameworks used in game development, such as Unity, Unreal Engine, and Game Maker Studio.
3)   Programming skills: Students should gain hands-on experience with programming languages commonly used in game development, such as C++, C#, or Java, and be able to apply them to create games.

4) Understanding of game architecture: Students should be able to understand and implement various game architectures, such as game loops, state machines, and event-driven programming.

5) Understanding of game AI: Students should be able to comprehend and implement various game AI techniques, such as pathfinding, decision-making, and behavior trees.

TYCS Semester VI

USCS601 Wireless Sensor Networks and Mobile Communication

1. Students will understand the basics of wireless communication, including wireless signal propagation, modulation, and coding techniques.
2. Students will be able to design, implement and deploy a wireless sensor network using various sensor nodes and communication protocols.
3. Students will learn about various mobile communication technologies such as GSM, CDMA, and LTE, and their applications in different settings.
4. Students will understand the basics of mobile network architecture and design, including cellular networks, ad-hoc networks, and mesh networks.

USCS602 Cloud Computing

2. Understand the fundamentals of cloud computing, including the different deployment models and service models.
3. Understand the various types of cloud computing platforms, such as Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS).
4. Understand the concepts of virtualization, scalability, elasticity, and resource pooling in cloud computing.
5. Analyze and evaluate cloud computing architectures, including public, private, and hybrid clouds, and understand the security and privacy concerns associated with each.
6. Understand the various cloud computing services, including storage, compute, and networking services, and their role in developing and deploying applications on the cloud.

USCS603 Information Retrieval

1. Understand the fundamental concepts and history of information retrieval and its components.
2. Learn the Boolean retrieval model and understand how it is used in information retrieval.
3. Understand the concept of dictionaries and tolerant retrieval and their importance in IR.
4. Learn how to evaluate information retrieval results and apply personalized search techniques.
5. Develop knowledge of collaborative filtering and content-based recommendation of documents and products.
6. Develop knowledge of cross-lingual retrieval techniques and challenges.
7. Understand the architecture of web search engines and their components.
8. Develop knowledge of the challenges in XML retrieval and the vector space model for XML retrieval.
9. Apply information retrieval concepts and techniques to real-world problems, such as building personalized search engines and addressing web spam.
10. Develop skills in programming and data analysis tools commonly used in information retrieval research and practice, such as Python, R, and Hadoop.

USCS604 Data Science

1. Demonstrate proficiency with statistical analysis of data.
2. Build and assess data-based models.
3. Execute statistical analyses with professional statistical software.
4. Demonstrate skill in data management.
5. Apply data science concepts and methods to solve problems in real-world contexts and communicate these solutions effectively.

USCS605 Ethical Hacking

1) **Understanding of hacking concepts:** Students should be able to comprehend and explain various hacking concepts, such as vulnerability scanning, system hacking, and social engineering.
2) **Knowledge of hacking tools:** Students should be familiar with different hacking tools and their uses, including but not limited to password cracking, network scanning, and vulnerability assessment tools.
3) **Understanding of ethical and legal issues:** Students should be able to differentiate between ethical and unethical hacking and understand the legal implications of hacking activities.
4) **Risk assessment and mitigation:** Students should be able to identify potential security risks and vulnerabilities in a system or network and develop strategies to mitigate them.
5) **Professionalism and ethics:** Students should understand the importance of ethical behavior, professionalism, and responsible conduct in the field of cybersecurity.
6) **Communication skills:** Students should be able to communicate effectively with stakeholders, including technical and non-technical audiences, about security risks, vulnerabilities, and mitigation strategies.
7) **Practical skills**: Students should gain hands-on experience with various hacking techniques and tools, including penetration technique.

JNIESTRT

Yashwantrao Chavan College of Arts, Commerce & Science
Koperkhairane NAVI MUMBAI
(Permitted by Govt. of Maharashtra, Affiliated to University of Mumbai)

Yashwantrao Chavan College of Arts, Commerce & Science
Koperkhairane NAVI MUMBAI
(Permitted by Govt. of Maharashtra, Affiliated to University of Mumbai)