Independent University, Bangladesh
Department of Computer Science and Engineering

Program Educational Objectives(PEO) for Department of CSE

PEO1: Think Critically

PEO2: Design Conscientiously

PEO3: Implement Efficiently

Program Learning Outcomes (PLO) for Department of CSE

The learning outcomes of the degree program support all the outcomes suggested in the ABET criteria.

1. **Knowledge:** An ability to select and apply the knowledge, techniques, skills, and modern tools of the computer science and engineering discipline;
2. **Requirement Analysis:** An ability to identify, analyze, and solve a problem by defining the computing requirements of the problem through effectively gathering of the actual requirements;
3. **Problem Analysis:** An ability to select and apply the knowledge of mathematics, science, engineering, and technology to computing problems that require the application of principles and applied procedures or methodologies;
4. **Design:** An ability to design computer based systems, components, or processes to meet the desire requirement;
5. **Problem Solving:** An ability to apply mathematical foundations, simulation, algorithmic principles, and computer science theory in the modeling and design of computer-based systems in a way that demonstrates comprehension of the tradeoffs involved in design choices.
6. **Implementation:** An ability to apply design and development principles in the construction of software systems of varying complexity
7. **Experiment and Analysis:** An ability to conduct standard tests and measurements; to conduct, analyze, and interpret experiments; and to apply experimental results to improve solutions (products or processes);
8. **Community Engagement and Engineering:** An ability to appreciate human behavior, culture, interaction and organization through studies in the humanities and social sciences. A knowledge of the impact of computing solutions in a local and global context;
9. **Teamwork:** An ability to function effectively as a member or leader of a technical team to accomplish common goals;
10. **Communication:** An ability to apply written and oral communication in both technical and non-technical environments; an ability to communicate with a range of audience; and an ability to identify and use appropriate available technical literature;

11. **Self-Motivated:** Recognition of the need for and an ability to engage in self-directed continuing professional development; prepared to enter a top-ranked graduate program in Computer Science and Engineering.
12. **Ethics:** An understanding of and a commitment to address professional, ethical, legal, security, social issues and responsibilities including a respect for diversity;
13. **Process Management:** A commitment to quality, timeliness, and continuous improvement.

PLO – PEO mapping

	PEO 1	**PEO 2**	**PEO 3**
PLO 1: Knowledge	✓	✓	✓
PLO 2: Requirement Analysis	✓	✓	
PLO 3: Problem Analysis	✓	✓	
PLO 4: Design	✓	✓	✓
PLO 5: Problem Solving	✓	✓	✓
PLO 6: Implementation		✓	✓
PLO 7: Experiment and Analysis	✓		✓
PLO 8: Community Engagement & Engg.		✓	✓
PLO 9: Teamwork		✓	✓
PLO 10: Communication	✓		✓
PLO 11: Self-Motivated		✓	
PLO 12: Ethics	✓	✓	✓
PLO 13: Process Management		✓	✓

CSC 101: Introduction to Computer Programming

Course Description

This is an introductory course in Computer Science. The main objective of this course is to help the student develop a strong foundation of computer programming using C++. The programming concepts that will be covered in the class are variable, data types, input, output, arithmetic operation, control structures, logical operation, conditional statements, iterative statements, array, function, string. Each lecture would involve solving a number of programming problems using the computer. After successful completion of the course a student will be able to break down a complex programming problem into smaller parts, solve them and write the solution in C++.

Course Outcomes

1. Know about different data types, operators and memory access techniques. Reason about interleaved statements operating on a shared data structure
2. Reason about compile errors, common runtime errors (e.g. NullPE) and logical errors in given short code segments (1-10 lines)
3. Reason about short-circuiting & different code paths for different data control structures and repeat structures
4. Know about procedural coding and in-line coding, direction and indirection operators, call by value and call by reference. Reason about computational cost, and return values
5. Competence in using an industry-standard fully-featured modern IDE (e.g. Visual Studio, CodeBlocks) as a development tool.
6. Know how to analyze and solve a problem formally.

CSC 101 PLO-CO mapping

	CO 1	CO 2	CO 3	CO 4	CO 5	CO 6
PLO 1: Knowledge	✓	✓				
PLO 2: Requirement Analysis	✓			✓		✓
PLO 3: Problem Analysis	✓	✓	✓	✓		✓
PLO 4: Design			✓	✓		
PLO 5: Problem Solving		✓				✓
PLO 6: Implementation					✓	✓
PLO 7: Experiment and Analysis		✓	✓			
PLO 8: Community Engagement & Engg.						
PLO 9: Teamwork					✓	
PLO 10: Communication						
PLO 11: Self-Motivated						
PLO 12: Ethics	✓					
PLO 13: Process Management					✓	✓

CSC 201: Discrete Mathematics

Course Description

This course covers elementary discrete mathematics required for computer science and engineering students. Emphasis is placed on mathematical definitions and proofs as well as methods of application. Topics include a review of set theory, formal logic notation and operations, methods of proof, induction, permutations and combinations, basic and advanced counting techniques, recurrence relations, generating functions, graph theory and finite state machines.

Course Outcomes

1. Understand concepts of counting, and sets. Understand different concepts of Bounds and Order (big-O, Omega and Theta) with running time for simple pseudo-code examples, especially recursive examples. Includes finding closed-forms for recursively-defined formulas using unrolling and recursion trees
2. Understand recurrence and recursive functions.
3. Know the basics of FOL (First Order Logic), Apply predicate logic: determine the truth of statements, perform simple transformations (esp. negation), accurately apply formal definitions (esp. vacuous truth cases, attention to variable types and scope)
4. Understand different proof techniques (Proof by Construction/Contradiction/Induction) and be able to apply them.
5. State and apply basic definitions, facts, and notation for commonly used discrete mathematics and graph theoretic constructs like graphs and trees.
6. Classify the complexity problem solving in terms of countable versus uncountable, polynomial versus exponential (P vs. NP), decidable versus undecidable. Know existence of different knowledge domains: Known, Unknown, Unknowable.

CSC201 PLO-CO mapping

	CO 1	CO 2	CO 3	CO 4	CO 5	CO 6
PLO 1: Knowledge	✓		✓		✓	✓
PLO 2: Requirement Analysis						
PLO 3: Problem Analysis	✓	✓				
PLO 4: Design	✓					
PLO 5: Problem Solving	✓	✓	✓	✓		
PLO 6: Implementation	✓		✓		✓	
PLO 7: Experiment and Analysis	✓		✓			✓
PLO 8: Community Engagement & Engg.						
PLO 9: Teamwork						
PLO 10: Communication				✓		✓
PLO 11: Self-Motivated				✓		
PLO 12: Ethics						
PLO 13: Process Management						

CSC 203: Data Structure

Course Description

This is an introductory Data Structure course in Computer Science and Engineering with particular emphasis on logic building and algorithm designing; able to communicate with computers through data structures that are designed using C++ programming language. The main objective of this course is to enable students thinking logically and rationally. Understanding basic concepts and theories of data structure would inspire students to build workable solutions for algorithmic or computer based computational problems.

Course Outcomes

1. Good understanding of dynamic memory allocation as opposed to Static memory allocation, difference between random memory access structures (Array) and pointer based memory access (Linked List). Be able to navigate, organize, and compile C++ projects of moderate complexity (many objects and dependencies).
2. Decompose a problem into its supporting data structures such as lists, stacks, queues, trees, etc.
3. Know different search techniques (BFS, DFS). To be able to decide on appropriate data structure to implement efficient algorithms. To be able to solve problems using techniques like graph search, tree traversal, optimization, data organization, etc.
4. Implement classic and adapted data structures and applications.
5. Analyze the efficiency of implementation choices.

CSC 203 PLO-CO mapping

	CO 1	CO 2	CO 3	CO 4	CO 5
PLO 1: Knowledge	✓	✓	✓		✓
PLO 2: Requirement Analysis	✓	✓	✓		✓
PLO 3: Problem Analysis		✓	✓		
PLO 4: Design	✓	✓	✓		
PLO 5: Problem Solving	✓			✓	
PLO 6: Implementation	✓			✓	
PLO 7: Experiment and Analysis		✓	✓		✓
PLO 8: Community Engagement & Engg.					
PLO 9: Teamwork	✓		✓		
PLO 10: Communication			✓		✓
PLO 11: Self-Motivated					
PLO 12: Ethics					✓
PLO 13: Process Management	✓				

CSC 204: Introduction to Hardware & Digital Logic Design

Course Description

This course provides a modern introduction to logic design and the basic building blocks used in digital systems, in particular digital computers. It starts with a discussion of combinational logic: logic gates, minimization techniques and arithmetic circuits. The second part of the course deals with sequential circuits: flip-flops, synthesis of sequential circuits, and case studies, including counters, registers, and random access memories. It provides coverage of classical hardware design for both combinational and sequential logic circuits. The course is supported by a digital logic design laboratory. State machines will then be discussed and illustrated through case studies of more complex problems. Different representations including truth table, logic gate, timing diagram, switch representation, and state diagram will be discussed.

Course Outcomes

1. Understand and be able to demonstrate fundamental concepts and techniques used in digital electronics.
2. Appreciate the structure of various number systems and its application in digital design.
3. Understand, analyze and design various combinational and sequential circuits.
4. Identify and analyze various hazards and timing problems in a digital design.
5. Build and analyze digital circuits.

CSC 204 PLO-CO mapping

	CO 1	CO 2	CO 3	CO 4	CO 5
PLO 1: Knowledge	✓	✓	✓	✓	
PLO 2: Requirement Analysis	✓	✓	✓	✓	
PLO 3: Problem Analysis	✓	✓	✓	✓	✓
PLO 4: Design	✓		✓		✓
PLO 5: Problem Solving	✓	✓	✓	✓	✓
PLO 6: Implementation			✓	✓	✓
PLO 7: Experiment and Analysis			✓	✓	✓
PLO 8: Community Engagement & Engg.					
PLO 9: Teamwork			✓	✓	✓
PLO 10: Communication		✓	✓	✓	✓
PLO 11: Self-Motivated					✓
PLO 12: Ethics		✓	✓	✓	
PLO 13: Process Management			✓	✓	✓

CSC 212: Microprocessor Interfacing and Assembly Language

Course Description

Introduction to the 80x86 families of microprocessors and the organization of an IBM PC. Topics covered: Microprocessor architecture, addressing mechanism, Instruction set, Instruction format; Assembly language programming: assembling, linking, running and debugging programs; Program control instructions and interrupts; Microprocessor interfacing with memory and other devices; 8086 based system design, Programmable peripheral interface: 8255A, 8251A, DMA controller 8237, Interrupt controller 8259A; Overview of advanced processors: 80386, Pentium and Multicore processors.

Course Outcomes

1. Analyze and understand bus/interface structures.
2. Characterize the timing/performance behavior of interfaces.
3. Utilize Assembly language programs to gain insight into instructions and machine-level operations.
4. Program and debug microprocessor devices.
5. Control/use peripherals, devices, and buses.

CSC212 PLO-CO mapping

	CO 1	CO 2	CO 3	CO 4	CO 5
PLO 1: Knowledge	✓		✓		
PLO 2: Requirement Analysis	✓		✓		
PLO 3: Problem Analysis		✓			
PLO 4: Design	✓				
PLO 5: Problem Solving				✓	
PLO 6: Implementation		✓	✓	✓	✓
PLO 7: Experiment and Analysis		✓	✓	✓	
PLO 8: Community Engagement & Engg.					✓
PLO 9: Teamwork			✓		✓
PLO 10: Communication					
PLO 11: Self-Motivated				✓	
PLO 12: Ethics					✓
PLO 13: Process Management					✓

CSC 301: Finite Automata and Computability

Course Description

This is a rigorous introduction to the core concepts of theoretical computer science. Through this course, students will learn to analyze and classify problems according to complexity and computability. Topics covered include fundamental concepts such as string, prefix, suffix, substring, concatenation; Cardinality; Distinction between uncountable and countable infinite; Different proof techniques: Proof by construction, proof by contradiction, pigeonhole principle; Deterministic and non-deterministic Finite state automata; Regular and non-regular languages, regular expressions; Equivalence of NFA and DFA; Pumping Lemma; Context free grammar (CFG) and Push down automata (PDA); Chomsky Normal form; Parsing; Turing machine, Universal Turing machine and the Halting problem; Goedel numbering; Computability (P vs NP).

Course Outcomes

1. Students will learn methods for designing efficient algorithms, evaluating their performance, and ways of establishing precise limits on the possible effectiveness of classes of algorithms
2. They will learn standard algorithms for fundamental problems.

CSC301 PLO-CO mapping

	CO 1	CO 2
PLO 1: Knowledge	✓	✓
PLO 2: Requirement Analysis	✓	✓
PLO 3: Problem Analysis	✓	✓
PLO 4: Design	✓	
PLO 5: Problem Solving	✓	✓
PLO 6: Implementation	✓	✓
PLO 7: Experiment and Analysis	✓	
PLO 8: Community Engagement & Engg.		
PLO 9: Teamwork		
PLO 10: Communication		
PLO 11: Self-Motivated		
PLO 12: Ethics	✓	
PLO 13: Process Management	✓	

CSC 305: Object Oriented Programming

Course Description

This is an advanced programming course in computer science, with particular emphasis on advanced concepts and theories of object oriented programming. The main objective of this course is to enable students designing software using object oriented approach. Understanding object oriented concepts and theories would inspire students in designing and developing desktop application software (both console and GUI based) using C++ and Java programming language.

Course Outcomes

1. Understand the benefits of object oriented design and when it is an appropriate methodology to use.
2. Design, write and test programs that make appropriate use of advanced object-oriented facilities common to object-oriented languages such as overloading and inheritance.
3. Manipulate classes provided in the programming API and incorporate them into solutions.

CSC305 PLO-CO mapping

	CO 1	CO 2	CO 3
PLO 1: Knowledge	✓		✓
PLO 2: Requirement Analysis	✓		
PLO 3: Problem Analysis	✓		
PLO 4: Design		✓	✓
PLO 5: Problem Solving		✓	
PLO 6: Implementation		✓	✓
PLO 7: Experiment and Analysis		✓	
PLO 8: Community Engagement & Engg.	✓		✓
PLO 9: Teamwork		✓	
PLO 10: Communication	✓		
PLO 11: Self-Motivated		✓	
PLO 12: Ethics	✓		
PLO 13: Process Management		✓	✓

CSC 306: Algorithm

Course Description

Algorithms are recipes for solving computational problems. In this course we will study fundamental algorithms for solving a variety of problems, including sorting, searching and graph algorithms. More importantly, we will focus on general design and analysis techniques that underlie these algorithms. For example, we will examine divide-and-conquer, dynamic programming, greediness, and probabilistic approaches. With an understanding of these techniques, we will be prepared to design some of our own algorithms.

Course Outcomes

1. Students will learn methods for designing efficient algorithms, evaluating their performance, and ways of establishing precise limits on the possible effectiveness of classes of algorithms
2. They will learn standard algorithms for fundamental problems.

CSC306 PLO-CO mapping

	CO 1	CO 2
PLO 1: Knowledge	✓	✓
PLO 2: Requirement Analysis	✓	
PLO 3: Problem Analysis	✓	✓
PLO 4: Design	✓	
PLO 5: Problem Solving	✓	✓
PLO 6: Implementation	✓	✓
PLO 7: Experiment and Analysis	✓	
PLO 8: Community Engagement & Engg.		
PLO 9: Teamwork		
PLO 10: Communication		
PLO 11: Self-Motivated		
PLO 12: Ethics	✓	
PLO 13: Process Management	✓	

CSC 311: Computer Organization and Architecture

Course Description

This is one of the core courses of Computer Science and Engineering with particular emphasis on computer organization and architecture; concept of computer as hierarchical system; and problems and methods of designing computers. The main objective of this course is to learn how certain operating system functions are supported by computer hardware organization. Understanding how various performance enhancements to computers are achieved and to be able to make an informed comparison among competing architectures for a given purpose.

Course Outcomes

1. Understand design principles and methods used in contemporary processors and memory systems and apply them to new designs.
2. Evaluate the performance of a modern computer.
3. Determine sources of potential performance bottlenecks in a processor design and determine techniques to address them.
4. Reason about sources of low memory system performance for a workload and determine techniques to address them.
5. Evaluate tradeoffs between hardware and software techniques to achieve a performance goal.

CSC311 PLO-CO mapping

	CO 1	CO 2	CO 3	CO 4	CO 5
PLO 1: Knowledge	✓				✓
PLO 2: Requirement Analysis	✓	✓			
PLO 3: Problem Analysis	✓		✓	✓	✓
PLO 4: Design	✓		✓		
PLO 5: Problem Solving	✓		✓	✓	✓
PLO 6: Implementation	✓				✓
PLO 7: Experiment and Analysis		✓	✓		✓
PLO 8: Community Engagement & Engg.					
PLO 9: Teamwork					
PLO 10: Communication				✓	
PLO 11: Self-Motivated					
PLO 12: Ethics		✓			
PLO 13: Process Management		✓			

CSC 317: Numerical Methods

Course Description

With the exponential increase in computing power numerical methods are becoming more and more relevant, useful and necessary for solving many important problems in diverse fields. The primary goal of this course is to provide a basic knowledge of different mathematical techniques to solve numerical problems that arise in many different fields of Engineering, Computer Science, and Physics. Topics covered in this course are the following: solutions of linear systems of equations using Elimination and Iterative methods, methods to obtain the eigenvalues and eigenvectors of a matrix, curve fitting and difference tables, numerical differentiation and integration using Newton's difference method and direct fit polynomials, solution of 1-dimensional ordinary differential equations. MATLAB software will be used for the implementation of these numerical techniques. The Lab segment of the course will be used to gain hands-on experience in numerical methods by coding in MATLAB.

Course Outcomes

1. Analyze the sources of errors in mathematical operations on the computer.
2. Recognize major numerical methods and their merits and pitfalls.
3. Calculate the computational cost of a range of numerical methods.
4. Select and use software tools, based on their numerical methods, for a range of problems.
5. Estimate the accuracy in approximated numerical solutions.

CSC317 PLO-CO mapping

	CO 1	CO 2	CO 3	CO 4	CO 5
PLO 1: Knowledge	✓	✓			
PLO 2: Requirement Analysis		✓			
PLO 3: Problem Analysis	✓	✓		✓	
PLO 4: Design					
PLO 5: Problem Solving				✓	✓
PLO 6: Implementation			✓	✓	
PLO 7: Experiment and Analysis			✓		✓
PLO 8: Community Engagement & Engg.					
PLO 9: Teamwork				✓	
PLO 10: Communication					
PLO 11: Self-Motivated				✓	
PLO 12: Ethics					
PLO 13: Process Management				✓	

CSC 401: Database Management

Course Description

Conventional and database approaches. Basic concepts of DBMS. Hierarchical, network and relational data models. Entity-relationship modeling. Relational database designing: decomposition and normalization; functional dependencies. Relational algebra and calculus. Structured query language (SQL). Query optimization. Database programming with SQL and PL/SQL. Database security and administration. Distributed databases. Object-oriented data modeling. Specific database systems: Oracle, MS SQL server, access.

Course Outcomes

1. Proficiency in the design of database applications starting from the conceptual design to the implementation of database schemas and user interfaces.
2. Solid foundation on database design concepts, data models (E/R model, relational model), the database query language SQL, and components of a database management system.
3. Basic understanding of data access structures, query processing and optimization techniques, and transaction management.

CSC401 PLO-CO mapping

	CO 1	CO 2	CO 3
PLO 1: Knowledge		✓	✓
PLO 2: Requirement Analysis	✓	✓	
PLO 3: Problem Analysis		✓	
PLO 4: Design	✓		✓
PLO 5: Problem Solving			✓
PLO 6: Implementation	✓		✓
PLO 7: Experiment and Analysis			✓
PLO 8: Community Engagement & Engg.			
PLO 9: Teamwork			
PLO 10: Communication			
PLO 11: Self-Motivated			
PLO 12: Ethics	✓		
PLO 13: Process Management	✓		

CSC 405: System Analysis and Design

Course Description

This course examines the tools and techniques used for the design and analysis of information systems. Topics covered include: Systems and models; Project management; Tools for determining system requirements; data flow diagrams; decision table and decision trees; Systems analysis: systems development life cycle models. Object oriented analysis: use-case modeling, Unified Modeling Language. Feasibility analysis, Structured analysis; systems prototyping; system design and implementation: application architecture, user interface design. Front-end and backend design; database design; software management and hardware selection. Case studies of Information Systems.

Course Outcomes

1. Understand the system development life cycle.
2. Be familiar with a variety of problem solving tools and approaches for the design and analysis of information systems.
3. Write System Requirements (functional and non-functional).
4. Draw context and level-0 data flow diagrams.
5. Decompose level-0 DFD to level-1 DFD.
6. Draw Entity Relationship diagrams to represent common business situations.

CSC405 PLO-CO mapping

	CO 1	CO 2	CO 3	CO 4	CO 5	CO 6
PLO 1: Knowledge	✓	✓				
PLO 2: Requirement Analysis		✓	✓	✓	✓	
PLO 3: Problem Analysis		✓		✓	✓	
PLO 4: Design	✓	✓		✓	✓	✓
PLO 5: Problem Solving		✓				✓
PLO 6: Implementation			✓	✓		
PLO 7: Experiment and Analysis						
PLO 8: Community Engagement & Engg.			✓			✓
PLO 9: Teamwork						✓
PLO 10: Communication			✓			✓
PLO 11: Self-Motivated						
PLO 12: Ethics			✓			
PLO 13: Process Management						

CSC 411: Compiler Construction

Course Description

The course aims to introduce the components of a compiler. A compiler is a large program that takes a source language and translates it for execution on some target machine. The course will cover in detail the front end of an compiler that deals with lexical analysis, syntax analysis, semantic analysis, and Intermediate code generation. The front end focuses on the understanding the source language program and is largely independent of the target machine architecture. We will make use of different front end tools such as Lex/Flex and Yacc/Bison to build a front end of a compiler. The course will also introduce the back end components of the compiler that focuses on code optimization and code generation. Advanced topics such as Instruction scheduling and Register allocation will also be covered.

Course Outcomes

1. Identify the dataflow problem(s) required for a given dataflow optimization.
2. Construct and solve the dataflow equations for a given dataflow problem.
3. Identify the classical optimizations that could be applicable to a given piece of code to improve its performance.
4. Analyze the major control flow properties of a program, including control flow graphs, dominators, natural loops, and reducible vs. irreducible flow graphs.
5. Translate a source-level language into a low-level compiler internal representation.
6. Choose the appropriate compiler internal representation for different kinds of compiler tasks.
7. Implement the major phases of a simple compiler, including scanning, parsing, intermediate code generation, and a few program optimizations.

CSC411 PLO-CO mapping

	CO 1	CO 2	CO 3	CO 4	CO 5	CO 6	CO 7
PLO 1: Knowledge	✓		✓				
PLO 2: Requirement Analysis			✓	✓		✓	✓
PLO 3: Problem Analysis	✓						
PLO 4: Design				✓			✓
PLO 5: Problem Solving		✓			✓		
PLO 6: Implementation					✓		
PLO 7: Experiment and Analysis		✓				✓	✓
PLO 8: Community Engagement & Engg.							
PLO 9: Teamwork							✓
PLO 10: Communication							
PLO 11: Self-Motivated						✓	
PLO 12: Ethics							
PLO 13: Process Management			✓				✓

CSC 413: Design of Operating System

Course Description

This is one of the core courses of Computer Science and Engineering with particular emphasis on operating system design. Operating systems are central to computing activities; acts as an intermediary between a user of a computer and the computer hardware. Two primary aims of operating systems are to manage resources (e.g. CPU time, memory, storage and Input/output) and to control users and software. Operating system design goals are often contradictory and vary depending on user, software, and hardware criteria. This course illustrates the fundamental concepts behind operating systems, and examines the ways that design goals can be achieved.

Course Outcomes

1. Explain and implement kernel programming principles.
2. Explain basic OS components and the inter-dependencies among operating system components such as process management, memory management, file system management, I/O management, as well as get understanding of implementation of some of these basic components through machine problems.
3. Explain, analyze and argue system tradeoffs based on OS design decisions.
4. Explain and analyze the performance impact of basic operating system concepts and principles on parallel/distributed OS, mobile OS, multimedia OS and cloud OS.
5. Explain, analyze, and argue about OS security issues and their impact on various OS components.

CSC413 PLO-CO mapping

	CO 1	CO 2	CO 3	CO 4	CO 5
PLO 1: Knowledge	✓	✓		✓	
PLO 2: Requirement Analysis		✓			
PLO 3: Problem Analysis		✓	✓	✓	✓
PLO 4: Design		✓	✓		
PLO 5: Problem Solving			✓	✓	
PLO 6: Implementation	✓				
PLO 7: Experiment and Analysis		✓			
PLO 8: Community Engagement & Engg.					✓
PLO 9: Teamwork			✓		
PLO 10: Communication	✓		✓		
PLO 11: Self-Motivated					
PLO 12: Ethics				✓	✓
PLO 13: Process Management				✓	✓

CSC 430: Data Communication and Computer Networks

Course Description

Basic concepts, categories of networks, network topologies, OSI model and TCP/IP protocol suite, TCP/IP applications, FTP, SMTP, HTTP and WWW, transport layer protocols, Internetworking devices, repeaters, bridges and routers, routing algorithms, IP addressing, sub netting, domain name systems, Network programming: Client-Server programming, socket programming, data link control protocols, LAN types and technology, MAC protocols, high speed LANs and Gigabit Ethernet, Wireless LANs, MAN, Circuit switching and Packet switching, ISDN, Frame Relay and ATM, SONET/SDH. Spectrum and bandwidth, Digital Transmission, encoding, modulations and demodulations, multiplexing: FDM, TDM and WDM, interfaces and modems, transmission media, fiber optic and wireless media, error detection techniques.

Course Outcomes

1. Identify the problems that arise in networked communication
2. Explain the advantages and disadvantages of existing solutions to these problems in the context of different networking regimes
3. Understand the implications of a given solution for performance in various networking regimes
4. Evaluate novel approaches to these problems
5. Identify and describe the purpose of each component of the TCP/IP protocol suite
6. Develop solid client-server applications using TCP/IP
7. Understand the impact of trends in network hardware on network software issues

CSC430 PLO-CO mapping

	CO 1	CO 2	CO 3	CO 4	CO 5	CO 6	CO 7
PLO 1: Knowledge	✓	✓			✓		✓
PLO 2: Requirement Analysis		✓					
PLO 3: Problem Analysis	✓	✓		✓			✓
PLO 4: Design			✓	✓			
PLO 5: Problem Solving			✓	✓			
PLO 6: Implementation			✓			✓	
PLO 7: Experiment and Analysis				✓			
PLO 8: Community Engagement & Engg.							✓
PLO 9: Teamwork						✓	
PLO 10: Communication		✓					
PLO 11: Self-Motivated				✓			
PLO 12: Ethics	✓						
PLO 13: Process Management						✓	

CSC 445: Software Engineering

Course Description

Introduction to the principles of software engineering; Software as product and process; Project management and planning; tracking and scheduling; risk analysis and quality assurance techniques; Configuration management. Project and process metrics, size and function oriented metrics. Software testing techniques: black box and white box techniques. Testing strategy: unit, integration and system testing. Concepts of object-oriented, event-driven and network programming, client-server architecture, web engineering. The course focuses on taking a group development project from beginning to end.

Course Outcomes

1. Explain and apply the main aspects of software engineering
2. Evaluate requirements for a software system and apply the process of analysis and design using the object-oriented approach.
3. Employ group working skills including general organization, planning and time management and inter-group negotiation.
4. Translate a requirements specification into an implementable design, following a structured and organized process.
5. Formulate a testing strategy for a software system, employing techniques such as unit testing, test driven development and functional testing.

CSC445 PLO-CO Mapping

	CO 1	CO 2	CO 3	CO 4	CO 5
PLO 1: Knowledge	✓	✓			
PLO 2: Requirement Analysis		✓		✓	
PLO 3: Problem Analysis		✓			
PLO 4: Design		✓		✓	
PLO 5: Problem Solving		✓			✓
PLO 6: Implementation				✓	✓
PLO 7: Experiment and Analysis					✓
PLO 8: Community Engagement & Engg.				✓	
PLO 9: Teamwork			✓		
PLO 10: Communication			✓		
PLO 11: Self-directed			✓		
PLO 12: Ethics			✓		
PLO 13: Process Management			✓	✓	✓

CSC 455: Web Applications and Internet

Course Description

This course serves as a comprehensive overview of web technologies and their usage. Essential topics such as OSI & TCP/IP architecture, Internet Routing, IP addressing & Domain Name System will be covered. Discussions will be held on popular browsers, HTML and Cascading Style Sheet, HTTP, HTTPS, FTP, Client and Server side scripts, Scripting (JavaScript, AJAX, XML) with jQuery libraries, Web Servers (IIS, Apache). Students will learn to design dynamic websites using ASP.NET with SQL server and PHP with My SQL. A brief overview of topics in web security such as cryptography, digital signatures, digital certificates, authentication & firewall will be provided.

Course Outcomes

1. Apply fundamental web-based client-server architecture concepts.
2. Write server-side programs to deliver dynamic content for web pages.
3. Create secure and scalable web applications.
4. Analyze and model requirements and constraints for the design of client-server internet applications.
5. Design and implement a client-server internet application that accommodates specific requirements and constraints, based on analysis, modeling or requirements specification.
6. Select and justify the implications of implementation and design considerations to internet security.

CSC455 PLO-CO mapping

	CO 1	CO 2	CO 3	CO 4	CO 5	CO 6
PLO 1: Knowledge	✓					
PLO 2: Requirement Analysis	✓			✓	✓	
PLO 3: Problem Analysis	✓			✓	✓	
PLO 4: Design	✓	✓		✓	✓	
PLO 5: Problem Solving	✓			✓		
PLO 6: Implementation		✓	✓		✓	
PLO 7: Experiment and Analysis		✓				✓
PLO 8: Community Engagement & Engg.		✓			✓	
PLO 9: Teamwork			✓			
PLO 10: Communication					✓	✓
PLO 11: Self-Motivated						
PLO 12: Ethics			✓			✓
PLO 13: Process Management			✓			

MAT203: Linear Algebra-Vector and Matrices

Course Description

The course Linear Algebra has two major components, matrix Algebra and vector spaces. Essentially Linear Algebra teaches how to deal with physical systems with very large number of governing independent variables. The relationship between the dependent and the independent variables is assumed to be linear- hence the name 'Linear Algebra'. It may appear at first sight that the assumption of linear dependency narrows the application of Linear Algebra, which is not true. To understand the nonlinear dependency in a physical system, it is first necessary to understand the linear dependency. The necessary mathematical tools, which are used to deal with a nonlinear system, is built upon the mathematical tools of a linear system. At present, the Linear Algebra forms the basis of most analysis in Physics, Engineering and many branches of Management Science. At this University, this course is required for students intending to major in computer science and Engineering.

Course Outcomes

1. Solve systems of linear equations using multiple methods, including Gaussian elimination and matrix inversion.
2. Carry out matrix operations, including inverses and determinants.
3. Demonstrate understanding of the concepts of vector space and subspace.
4. Demonstrate understanding of linear independence, span, and basis.
5. Determine eigenvalues and eigenvectors and solve eigenvalue problems.
6. Apply principles of matrix algebra to linear transformations.
7. Demonstrate understanding of inner products and associated norms.]

MAT203 PLO-CO mapping

	CO 1	CO 2	CO 3	CO 4	CO 5	CO 6	CO 7
PLO 1: Knowledge	✓	✓	✓	✓		✓	✓
PLO 2: Requirement Analysis							
PLO 3: Problem Analysis					✓		
PLO 4: Design						✓	
PLO 5: Problem Solving	✓				✓	✓	
PLO 6: Implementation	✓				✓	✓	
PLO 7: Experiment and Analysis							
PLO 8: Community Engagement & Engg.							
PLO 9: Teamwork							
PLO 10: Communication							
PLO 11: Self-Motivated							
PLO 12: Ethics							
PLO 13: Process Management							

MAT301: Ordinary Differential Equations

Course Description

The first part of the course is a continuation of calculus. We will cover optimization, integration techniques and partial derivatives. The second part of the course deals with ordinary differential equations. Differential equations are an indispensable tool for engineers and scientists. We will cover the most commonly used techniques for solving first order and second order equations. We will also cover some significant applications.

Course Outcomes

1. Identify, analyze and subsequently solve physical situations whose behavior can be described by ordinary differential equations.
2. Determine solutions to first order separable differential equations.
3. Determine solutions to first order linear differential equations.
4. Determine solutions to first order exact differential equations.
5. Determine solutions to second order linear homogeneous differential equations with constant coefficients.
6. Determine solutions to second order linear non-homogeneous differential equations with constant coefficients.

MAT301 PLO-CO mapping

	CO 1	CO 2	CO 3	CO 4	CO 5	CO 6
PLO 1: Knowledge	✓	✓	✓	✓	✓	✓
PLO 2: Requirement Analysis	✓	✓	✓	✓	✓	✓
PLO 3: Problem Analysis	✓	✓	✓	✓	✓	✓
PLO 4: Design	✓					
PLO 5: Problem Solving	✓	✓	✓	✓	✓	✓
PLO 6: Implementation	✓	✓	✓	✓	✓	✓
PLO 7: Experiment and Analysis	✓					
PLO 8: Community Engagement & Engg.						
PLO 9: Teamwork						
PLO 10: Communication						
PLO 11: Self-Motivated						
PLO 12: Ethics						
PLO 13: Process Management						

CSC 417: Data Mining and Warehouse

Course Description

Basic concept of data mining, issues and techniques. Data warehouse and OLTP technologies for data mining, Classification of data mining techniques and models, data pre-processing, data mining primitives, query languages and system architecture, characterization and comparison. Mining association rules in large database. Cluster analysis, multidimensional analysis and descriptive mining of complex data object. Data mining in distributed heterogeneous database systems. Data mining applications and future research issues.

Course Outcomes

1. Understand the basic principles for data cleaning and data transformation and apply typical methods of data cleaning and transformation in the context of data mining.
2. Understand the basic principles of data warehousing and data cubing and apply typical methods of data warehousing and data cube computation.
3. Understand the basic principles for mining frequent patterns and apply typical frequent pattern mining methods for effective data mining.
4. Understand the basic principles for classification and apply typical classification methods for effective data mining.
5. Understand the basic principles for data clustering and apply typical clustering methods for effective data mining.

CSC417 PLO-CO Mapping

	CO 1	CO 2	CO 3	CO 4	CO 5
PLO 1: Knowledge	✓	✓	✓	✓	✓
PLO 2: Requirement Analysis	✓	✓	✓	✓	✓
PLO 3: Problem Analysis	✓	✓	✓	✓	✓
PLO 4: Design	✓	✓	✓	✓	✓
PLO 5: Problem Solving	✓	✓	✓	✓	✓
PLO 6: Implementation	✓	✓	✓	✓	✓
PLO 7: Experiment and Analysis	✓	✓	✓	✓	✓
PLO 8: Community Engagement & Engg.					
PLO 9: Teamwork					
PLO 10: Communication					
PLO 11: Self-directed					
PLO 12: Ethics					
PLO 13: Process Management					

CSC 420: Image Processing

Course Description

This Course Introduces student's preliminaries of developing artificial vision based applications. This is the first curse on computer vision. In first part students learn low level vision. That would include: Image formation and Imaging Operations, Basic Image Filtering Operations, Thresholding Techniques, Edge Detection, Corner and Interest Point Detection, Mathematical Morphology, Texture. Students learn fundamentals of image formation in the digital environment, representation of images in digital environment, and, preprocessing, processing and post-processing of acquired image signals. This is a very practical course and taken in a lab environment all concept learned in the class will be implemented using MATLAB or OpenCV.

Course Outcomes

1. Understand the relevant aspects of digital image representation and their practical implications.
2. Have the ability to design pointwise intensity transformations to meet stated specifications.
3. Understand 2-D convolution, the 2-D DFT, and have the ability to design systems using these concepts.
4. Have a command of basic image restoration techniques.
5. Appreciate the utility of wavelet decompositions and their role in image processing systems.
6. Have an understanding of the underlying mechanisms of image compression.

CSC420 PLO-CO Mapping

	CO 1	CO 2	CO 3	CO 4	CO 5	CO 6
PLO 1: Knowledge	✓		✓	✓	✓	✓
PLO 2: Requirement Analysis	✓	✓			✓	
PLO 3: Problem Analysis	✓	✓				
PLO 4: Design		✓	✓			✓
PLO 5: Problem Solving		✓		✓	✓	
PLO 6: Implementation		✓	✓	✓		
PLO 7: Experiment and Analysis						
PLO 8: Community Engagement & Engg.						
PLO 9: Teamwork						
PLO 10: Communication						
PLO 11: Self-directed						
PLO 12: Ethics						
PLO 13: Process Management				✓		✓

CSC 421: Machine Learning

Course Description

The primary focus of this course is to learn techniques to make sense of numerical data. Ways to learn the hidden rules in raw data by reducing the size of data or refining the numerical data is taught in this course. Application of the methods learned in this course in finance, sociology, computer vision, psychology and other fields is also discussed. Major topics learned in this course includes: PCA, ICA, LDA, Decision Tree, clustering, ANN, HMM, SVM, Genetic algorithm and varied versions of these methods.

Course Outcomes

1. Be able to articulate key concepts and principles in Machine learning.
2. Be able to articulate and model problems given an understating of representational issues and abstraction in machine learning.
3. Be able to explain and analyze models and results making use of theoretical principles and the limitations of generalization in machine learning.
4. Make use of the algorithmic theory of machine learning in problem analysis and model selection.
5. Understand and apply the maximum likelihood principle and explain algorithmic implications in modeling and problem solving.
6. Be able to use a variety of algorithmic techniques in machine learning.
7. Be able to choose and use a variety of machine learning protocols in different situations.

CSC421 PLO-CO Mapping

	CO 1	CO 2	CO 3	CO 4	CO 5	CO 6	CO 7
PLO 1: Knowledge	✓	✓	✓		✓		✓
PLO 2: Requirement Analysis		✓		✓			
PLO 3: Problem Analysis		✓	✓	✓			
PLO 4: Design		✓		✓	✓		
PLO 5: Problem Solving			✓	✓	✓	✓	
PLO 6: Implementation				✓	✓	✓	✓
PLO 7: Experiment and Analysis			✓				✓
PLO 8: Community Engagement & Engg.							
PLO 9: Teamwork							
PLO 10: Communication							
PLO 11: Self-directed						✓	
PLO 12: Ethics							
PLO 13: Process Management							✓

CSE 424: Neural Networks

Course Description

This course aims to introduce the concepts of learning in artificial neural networks. The similarity and dissimilarities between natural and artificial neural networks is discussed in this course. The main objective of this course is to give the ideas to the learners on how to use artificial neural networks in real life problems. More specifically it will address the questions of: which types of problems are more suitable for artificial neural networks and which type of artificial neural network should be used for what type of problem.

Course Outcomes

1. Describe the relation between real brains and simple artificial neural network models.
2. Explain and contrast the most common architectures and learning algorithms for multilayer perceptrons, radial-basis function networks, committee machines and Kohonen self-organizing maps.
3. Discuss the main factors involved in achieving good learning and generalization performance in neural network systems.
4. Identify the main implementation issues for common neural network systems.
5. Evaluate the practical considerations in applying neural networks to real classification and regression problems.

CSC424 PLO-CO Mapping

	CO 1	CO 2	CO 3	CO 4	CO 5
PLO 1: Knowledge	✓	✓	✓		✓
PLO 2: Requirement Analysis		✓			
PLO 3: Problem Analysis		✓		✓	
PLO 4: Design		✓			
PLO 5: Problem Solving			✓	✓	
PLO 6: Implementation			✓	✓	
PLO 7: Experiment and Analysis					✓
PLO 8: Community Engagement & Engg.					✓
PLO 9: Teamwork					
PLO 10: Communication					
PLO 11: Self-directed					
PLO 12: Ethics					✓
PLO 13: Process Management			✓	✓	

CSE 425: Artificial Intelligence

Course Description

This is an advanced knowledge representation and machine intelligence course in Computer Science and Engineering with particular emphasis on problem solving techniques; able to command a computer through advanced algorithms based on human like reasoning. The main objective of this course is to teach students how to transfer intelligence into a machine through some computer's language. Understanding advanced concepts and theories of artificial intelligence would inspire students to build workable smart solutions of a computational problem.

Course Outcomes

1. Apply artificial intelligence techniques, including search heuristics, knowledge representation, planning and reasoning
2. Describe the key components of the artificial intelligence (AI) field
3. Explain search strategies
4. Solve problems by applying a suitable search method
5. Analyze and apply knowledge representation
6. Describe and list the key aspects of planning in artificial intelligence
7. Analyze and apply probability theorem and Bayesian networks
8. Describe the key aspects of intelligent agents

CSC425 PLO-CO Mapping

	CO 1	CO 2	CO 3	CO 4	CO 5	CO 6	CO 7	CO 8
PLO 1: Knowledge	✓	✓				✓		✓
PLO 2: Requirement Analysis	✓		✓			✓		
PLO 3: Problem Analysis	✓		✓			✓		
PLO 4: Design	✓		✓					
PLO 5: Problem Solving	✓			✓			✓	
PLO 6: Implementation	✓			✓			✓	
PLO 7: Experiment and Analysis	✓			✓	✓		✓	
PLO 8: Community Engagement & Engg.					✓			
PLO 9: Teamwork				✓				
PLO 10: Communication				✓				
PLO 11: Self-directed				✓				
PLO 12: Ethics					✓			
PLO 13: Process Management								

CSC 426: Introduction to Robotics

Course Description

The study of robotics includes many issues which are traditionally part of the computing sciences; distributed and adaptive control, architecture, software engineering, real-time systems, information processing and learning, mechanics and dynamics, geometrical reasoning, and artificial intelligence. Processing and mechanical functions of robots are dependent on the target platform and the world in which it is situated. A designer of an embedded computational system for sensory and motor processes needs to appreciate and understand all of these disciplines. Introduction to Robotics course is concerned with the design and analysis of basic robots. The focus will be on sensory and motor systems that interpret and manipulate their environments. In addition, we will study kinematics and dynamics, actuators, sensors, signal processing, associative memory, feedback control theory, supervised and unsupervised learning, and task planning.

Course Outcomes

1. Be able to analyze different robot motion systems and their errors.
2. Identify sensors and actuators required for specific applications.
3. Understand principles of mobile robots.
4. Understand programming principles for robot control.
5. Implement hardware and software to build a robot that can perform a task.

CSC426 PLO-CO Mapping

	CO 1	CO 2	CO 3	CO 4	CO 5
PLO 1: Knowledge	✓	✓	✓	✓	
PLO 2: Requirement Analysis		✓			✓
PLO 3: Problem Analysis		✓			✓
PLO 4: Design	✓			✓	✓
PLO 5: Problem Solving		✓			✓
PLO 6: Implementation				✓	✓
PLO 7: Experiment and Analysis	✓				✓
PLO 8: Community Engagement & Engg.					
PLO 9: Teamwork					✓
PLO 10: Communication					
PLO 11: Self-directed					
PLO 12: Ethics					
PLO 13: Process Management					✓

CSC 450: Cryptography and Network Security

Course Description

In a world dominated by digital communications and e-commerce, the need for security, confidentiality and integrity is paramount. In this course we will investigate different encryption and decryption techniques along with their pros and cons. We will also learn the current state of the art protocols and standards that are in practice. Finally we will explore the future of cryptography in a world on the brink of the advent of quantum computing.

Course Outcomes

1. Understand the principles and practices of cryptographic techniques.
2. Understand a variety of generic security threats and vulnerabilities, and identify and analyze particular security problems for a given application.
3. Appreciate the application of security techniques and technologies in solving real-life security problems in practical systems.
4. Design security protocols and methods to solve specified security problems.

CSC450 PLO-CO Mapping

	CO 1	CO 2	CO 3	CO 4
PLO 1: Knowledge	✓	✓	✓	
PLO 2: Requirement Analysis	✓	✓		
PLO 3: Problem Analysis	✓	✓		
PLO 4: Design	✓			✓
PLO 5: Problem Solving	✓		✓	✓
PLO 6: Implementation			✓	✓
PLO 7: Experiment and Analysis		✓		
PLO 8: Community Engagement & Engg.			✓	
PLO 9: Teamwork				✓
PLO 10: Communication				✓
PLO 11: Self-directed				
PLO 12: Ethics		✓		✓
PLO 13: Process Management				✓

CSC 452: Software Marketing

Course Description

Introduction to marketing and marketing communication, service marketing, marketing challenges of technology products and services; methodology to identify target buyer's perceptions and behaviors; marketing software product: market research, product positioning, determining value proposition of the product, pricing, distribution, promotions, intellectual property management, advertising and product management; marketing customized software development services: target market determination, educating clients, response to request for proposal, basics of software effort and cost estimation, client relationship management, business value determination of software solutions, base line data collection, determining return on software investment (ROI), impact assessment due to software use, and organization change management; marketing software engineering services to offshore clients: determining offshore market opportunities, understanding constraints in penetrating those markets, developing strategies, establishing linkages with complementary partners and packaging and promoting software engineering service capability to targeted markets; Software quality assurance: basics of software quality assurance (SQA), business value of SQA, and clients role in SQA. Introduction to ISO and SEI's capability maturity model for software SQA. Case studies. Contractual and legal issues.

Course Outcomes

1. Appreciate the global nature of software marketing and take appropriate measures to operate effectively in international settings.
2. Develop marketing strategies based on product, price, place and promotion objectives.
3. Comprehend the social, legal, ethical and technological forces behind software marketing decision-making.
4. Formulate software marketing strategies that incorporate psychological and sociological factors which influence consumers.
5. Understand and use the standardized models for software quality assurance.

CSC452 PLO-CO Mapping

	CO 1	CO 2	CO 3	CO 4	CO 5
PLO 1: Knowledge	✓		✓		✓
PLO 2: Requirement Analysis	✓	✓			
PLO 3: Problem Analysis	✓	✓			
PLO 4: Design		✓		✓	
PLO 5: Problem Solving		✓		✓	✓
PLO 6: Implementation		✓		✓	✓
PLO 7: Experiment and Analysis				✓	
PLO 8: Community Engagement & Engg.			✓	✓	
PLO 9: Teamwork					
PLO 10: Communication			✓		
PLO 11: Self-directed					
PLO 12: Ethics			✓	✓	
PLO 13: Process Management		✓			

CSC 454: Software Engineering Process Management

Course Description

Challenge of producing and maintaining complex software-intensive system; predictability and improved cost; team work in software development; quality assurance; process centric software engineering practices; software engineering process framework developed by Software Engineering Institute (SEI); capability of each process area; framework to meet challenges; characteristics of software products and processes, its quantification, analysis, prediction, control, and guidelines to achieve both business and technical goals.

Course Outcomes

1. Cooperate in and contribute to a team environment, develop team dynamics, work according to an agreed team protocol, and resolve/manage conflict issues.
2. Identify, analyze, compare and contrast different processes and their assistive tools for selected phases of the software engineering life cycle.
3. Utilize various software engineering processes and their tools as required for best-practice development of software systems.
4. Plan, identify and apply processes, standards and tools for phases of a software engineering life cycle for a substantial software development project.

CSC454 PLO-CO Mapping

	CO 1	CO 2	CO 3	CO 4
PLO 1: Knowledge		✓		
PLO 2: Requirement Analysis		✓		✓
PLO 3: Problem Analysis		✓		✓
PLO 4: Design			✓	✓
PLO 5: Problem Solving			✓	
PLO 6: Implementation			✓	✓
PLO 7: Experiment and Analysis		✓		
PLO 8: Community Engagement & Engg.		✓		
PLO 9: Teamwork	✓			✓
PLO 10: Communication	✓			
PLO 11: Self-directed	✓			
PLO 12: Ethics	✓		✓	
PLO 13: Process Management	✓			✓

CSC 458: Software Quality and Testing

Course Description

Software quality assurance (SQA), review of SQA practices, quality management, the role of SQA, the SQA program planning, launching and management, independent verification and validation; software inspections, basic principles, reviews, reporting and tracking, managing inspections and reviews; principles of software testing, testing types, test planning, development, execution and reporting; real-time testing and test organization; basic concepts of reliability, modeling software reliability from test results, techniques for analyzing, predicting, designing, and engineering the required and expected reliability of software systems.

Course Outcomes

1. Manage incidents and risks within a project.
2. Create test strategies and plans, design test cases, prioritize and execute them.
3. Apply modern software testing processes in relation to software development and project management.
4. Contribute to efficient delivery of software solutions and implement improvements in the software development processes.

CSC458 PLO-CO Mapping

	CO 1	CO 2	CO 3	CO 4
PLO 1: Knowledge			✓	
PLO 2: Requirement Analysis		✓		
PLO 3: Problem Analysis		✓		
PLO 4: Design		✓		
PLO 5: Problem Solving		✓	✓	
PLO 6: Implementation		✓	✓	✓
PLO 7: Experiment and Analysis		✓	✓	
PLO 8: Community Engagement & Engg.			✓	
PLO 9: Teamwork	✓			
PLO 10: Communication	✓			
PLO 11: Self-directed	✓			
PLO 12: Ethics	✓			✓
PLO 13: Process Management	✓		✓	✓

CSC 459: Software Architecture and Component-Based Design

Course Description

High-level architectural designs of software systems and products, strengths and weaknesses of each design style, component-based design, cohesion, interconnection and complexity, middleware, performance analysis and simulation, and COTS components; commonly-used software system structures, techniques for designing and implementing these structures, models and formal notations for characterizing and reasoning about architectures, tools for generating specific instances of an architecture, and case studies of actual system architectures Object-oriented design, design patterns, and UML; application of domain analysis, impact of platform dependence and independence, relation of software architecture to requirements, domain analysis and the architectural design process, and products in a real-world context.

Course Outcomes

1. Use well-understood paradigms for designing new systems.
2. Generate architectural alternatives for a problem and selection among them.
3. Describe a software architecture using various documentation approaches and architectural description languages.
4. Recognize major software architectural styles, design patterns, and frameworks.
5. Design and motivate software architecture for large-scale software systems.
6. Argue the importance and role of software architecture in large-scale software systems.
7. Discuss and evaluate the current trends and technologies such as model-driven and service-oriented architectures.

CSC459 PLO-CO Mapping

	CO 1	CO 2	CO 3	CO 4	CO 5	CO 6	CO 7
PLO 1: Knowledge	✓		✓	✓		✓	✓
PLO 2: Requirement Analysis	✓	✓			✓		✓
PLO 3: Problem Analysis		✓			✓		✓
PLO 4: Design		✓	✓	✓	✓		
PLO 5: Problem Solving		✓			✓		
PLO 6: Implementation	✓	✓			✓		
PLO 7: Experiment and Analysis		✓					✓
PLO 8: Community Engagement & Engg.						✓	
PLO 9: Teamwork		✓					
PLO 10: Communication			✓			✓	
PLO 11: Self-directed			✓				
PLO 12: Ethics						✓	
PLO 13: Process Management	✓					✓	