COMPUTER ENGINEERING

Program Outcome Computer Engineering

PO 1: Adequately disseminate the computing concepts and solutions to bridge the gap between computing industry experts and business leaders to create and initiate innovation.
PO 2: Identify, formulate, and solve Software Engineering, Networking and Data Mining problems.
PO 3: Effectively utilizing their knowledge of computing principles and mathematical theory to develop sustainable solutions to current and future computing problems
PO 4: Developing and implementing solution based systems and/or processes that address issues and/or improve existing systems within in a computing based industry
PO 5: Generate solutions by conducting experiments and applying techniques to analyze and interpret data

Course Outcome Computer Engineering

ECS- 301 Data Structure using C
CO 1: Understand the concept of Dynamic memory management, data types, algorithms, Big O notation.
CO 2: Understand basic data structures such as arrays, linked lists, stacks and queues.
CO 3: Describe the hash function and concepts of collision and its resolution methods
CO 5: Solve problem involving graphs, trees and heaps
CO 6: Apply Algorithm for solving problems like sorting, searching, insertion and deletion of data

ECS- 302   Discrete Structure and Graph theory

CO 1: Perform operations on various discrete structures such as sets, functions, relations, and sequences.
CO 2: Ability to solve problems using Counting techniques, Permutation and Combination, Recursion and generating functions.
CO 3: Apply algorithms and use of graphs and trees as tools to visualize and simplify Problems.
CO 4: Use of K-Maps and Truth Tables to construct and verify correctness of a Boolean expression.
CO 5: Understand the various properties of algebraic systems like Rings, Monoids and Groups.

ECS- 309 Digital logic Design

CO 1: Apply the principles of number system, binary codes and Boolean algebra to minimize logic expressions

CO 2: Develop K-maps to minimize and optimize logic functions up to 5 variables

CO 3: Acquire knowledge about various logic gates and logic families and analyze basic circuits of these families.

CO 4: Design various combinational and sequential circuits such as encoders , decoders and counters using multiplexers, and flip - flops

CO 5: Describe and compare various memory systems, shift registers and analog to digital and digital to analog conversion circuits

ECS- 401 Operating System

CO 1: Understand the basics of operating systems like kernel, shell, types and views of operating systems

CO 2: Describe the various CPU scheduling algorithms and remove deadlocks.

CO 3: Explain various memory management techniques and concept of thrashing

CO4: Use disk management and disk scheduling algorithms for better utilization of external memory.

CO 5: Recognize file system interface, protection and security mechanisms.

CO 6: Explain the various features of distributed OS like Unix, Linux, windows etc.

ECS-402 Theory of Automata and Formal Language

CO 1: Understand the basic concepts of formal languages, automata and grammar types, as well as the use of formal languages and reduction in normal forms

CO 2: Demonstrate the relation between regular expressions, automata, languages and grammar with formal mathematical methods

CO 3: Design push down automata, cellular automata and turing machines performing tasks of moderate complexity

 CO 4: Analyze the syntax and formal properties, parsing of various grammars such as LL(k) and LR(k)

CO 5: Describe the rewriting systems and derivation languages

ECS-403 Computer Graphics

CO 1: Understand the basics of computer graphics, different graphics systems and applications of computer graphics.
CO 2: Discuss various algorithms for scan conversion and filling of basic objects and their comparative analysis.
CO 3: Use of geometric transformations on graphics objects and their application in composite form. d) Extract scene with different clipping methods and its transformation to graphics display device.
CO 4: Explore projections and visible surface detection techniques for display of 3D scene on 2D screen.

ECS-409 Microprocessor and its application
CO 1: Understand the taxonomy of microprocessors and knowledge of contemporary microprocessors.
CO 2: Describe the architecture, bus structure and memory organization of 8085 as well as higher order microprocessors.
CO 3: Explore techniques for interfacing I/O devices to the microprocessor 8085 including several specific standard I/O devices such as 8251 and 8255.
CO 4: Demonstrate programming using the various addressing modes and instruction set of 8085 microprocessor
CO 5: Design structured, well commented , understandable assembly language programs to provide solutions to real world control problems

ECS-501 Design Algorithm and Analysis
CO 1: Ability to analyze the performance of algorithms.
CO 2: Ability to choose appropriate algorithm design techniques for solving problems.
CO3: Ability to understand how the choice of data structures and the algorithm design methods impact the performance of programs.
CO4: To clear up troubles the usage of a set of rules design methods including the grasping approach, divide and overcome, dynamic programming, backtracking and department and certain.
CO5: To understand the variations among tractable and intractable problems. To introduce p and np classes.

ECS - 502 Database Management System
CO 1: Define database system concepts and apply normalization to the database.
CO 2: Explain the basic processing and optimization techniques for high level query.
CO 3: Describe different transaction processing concepts and use different concurrency control techniques.
CO 4: Discuss different types of databases such as object oriented and distributed databases.
CO 5: Identify different types of database failures and techniques to recover from such failures.

CO 6: Discuss advanced database technologies and products used in enterprise.

ECS -505 Computer Architecture

CO 1:- Design basic and intermediate RISC pipelines, including the instruction set, data paths, and ways of dealing with pipeline hazards.

CO 2:-Consider various techniques of instruction-level parallelism, including superscalar execution, branch prediction, and speculation, in design of high-performance processors.

CO 3:- State and understand memory hierarchy design, memory access time formula, performance improvement techniques, and trade-offs.

CO 4:-State and compare properties of shared memory and distributed multiprocessor systems and cache memory.

ECS-504 Web Technology

CO 1: Explain the history of the internet and related internet concepts that are vital in understanding web development

CO 2: Discuss the insights of internet programming and implement complete application over the web

CO 3: Demonstrate the important HTML tags for designing static pages and separate design from content using Cascading Style sheet.

CO 4: Use web application development software tools i.e. Ajax, PHP and XML etc. and identify the environments currently available on the market to design web sites.

CO 5: Utilize the concepts of JavaScript and Java

ECS-601 Computer Networks

CO 1: Understand computer network basics, network architecture, TCP/IP and OSI reference models. Identify and understand various techniques and modes of transmission

CO 2: Describe data link protocols, multi-channel access protocols and IEEE 802 standards for LAN

CO 3: Describe routing and congestion in network layer with routing algorithms and classify IPV4 addressing scheme

CO 4: Discuss the elements and protocols of transport layer

CO 5: Understand network security and define various protocols such as FTP, HTTP, Telnet, DNS

ECS -603 Compiler Design

CO 1: Specify and analyse the lexical, syntactic and semantic structures of advanced language features.

CO 2: Separate the lexical, syntactic and semantic analysis into meaningful phases for a compiler to undertake language translation

CO 3: Write a scanner, parser, and semantic analyser without the aid of automatic generators

CO 4: Turn fully processed source code for a novel language into machine code for a novel computer

CO 5: Describe techniques for intermediate code and machine code optimisation

CO 6: Design the structures and support required for compiling advanced language features.

ECS- 602   Software Engineering

CO 1: Study a body of knowledge relating to Software Engineering, Software reengineering, and maintenance.

CO 2: Have skills in the most widely used approach to software construction – objectorientation (OO), including OO requirement specifications, OO analysis, OO design, OO • Programming, OO testing and maintenance.

CO 3: Develop an appreciation of the cost, quality, and management issues involved in software construction

CO4: Acquire skills to think about problems and their solutions using appropriate methods of analysis and design

ECS-604 Object Oriented Technique

CO 1: Describe the procedural and object oriented paradigm with concepts of streams, classes, functions, data and objects.

CO 2: Understand dynamic memory management technique using pointers, constructors, destructors, etc

CO 3: Describe the concept of function overloading, operator overloading, virtual functions and polymorphism.

CO 4: Classify inheritance with the understanding of early and late binding, usage of exception handling, generic programming**.**

CO 5: Demonstrate the use of various OOPs concepts with the help of programs.

Program Specific Outcome Computer Engineering

PSO 1: Ability to understand the principles and development methodologies of computer systems. Students can assess the hardware of computer systems and possess professional skills and knowledge of software design process

PSO 2: Ability to apply mathematical methodologies to solve computation tasks, model real world problems using appropriate data structure and suitable algorithms.

PSO 3:Ability to apply mathematical methodologies to solve computation task, model real world problem using appropriate data structure and suitable algorithm

PSO 4: Develop system solutions involving both hardware and software modules