SARASWATI Education Society's

SARASWATI College of Engineering

Learn Live Achieve and Contribute

Kharghar, Navi Mumbai - 410 210.

Course Objectives and Course Outcomes

Odd Semester

Class: SE

Subject code:**CSC302**	Subject: **Digital Logic Design and Analysis**	Credits:04

Course Objective:

At the end of course, student should be able to:

1	Identify the fundamental concepts for design of digital circuits.
2	Develop the methods for designing of digital circuits and a pre-requisite for computer organization and architecture, microprocessor systems.
3	Use the concept of designing Combinational.
4	Explain the concept of designing sequential circuits.
5	Outline the basic knowledge of how digital building blocks are described in VHDL.
6	Design the truth table for logical gates.

Course Outcomes:

At the end of course, students will attain an ability to:

1	Classify different number systems and their conversions.
2	Distinguish and minimize Boolean expressions.
3	Design and analyze combinational circuits.
4	Design and analyze sequential circuits
5	Illustrate the basic concepts of VHDL.
6	Justify basics of TTL and CMOS Logic families.

Subject code:**CSC303**	Subject: **Discrete Mathematics**	Credits:04

Course Objective:

At the end of course, student should be able to:

1	Discuss discrete Mathematics Concepts.
2	Demonstrate clear thinking and creative problem solving.
3	Construction and understanding of mathematical proofs. Exercise common mathematical and proof strategies.
4	Prepare for the mathematical aspects of other Computer Engineering courses
5	Develop the Technique for detecting and correcting code in transmitted data.
6	Discuss the importance of discrete structures towards simulation of a problem in computer science.

Course Outcomes:

At the end of course, students will attain an ability to:

1	Explain the notion of mathematical thinking, mathematical proofs and to apply them solving.
2	Illustrate reason logically.
3	Infer relations, Diagraph and lattice.
4	Demonstrate the use of functions, graphs and their use in programming applications.
5	Illustrate the use of groups and codes in Encoding-Decoding
6	Use discrete structures into other computing problems such as formal specification, artificial intelligence, cryptography, Data Analysis and Data Mining etc.

Subject code:**CSC305**	Subject :**Data Structure**	Credits:04

Course Objective:

At the end of course, student should be able to:

1	Define the fundamental concept of data structures and emphasize importance of Data Structures in developing and implementing efficient algorithms.
2	Explain efficient storage mechanisms of data for an easy access.
3	Design and demonstrate of various basic and advanced data structures.
4	Develop various techniques for representation of the data in the real world.
5	Discuss different sorting and searching techniques.
6	Extend the logical ability and understand the generic principles of computer programming as applied to sophisticated data structures.

Course Outcomes:

At the end of course, students will attain an ability to:

1	Explain various linear and nonlinear data structures.
2	Demonstrate operations like searching, insertion, deletion, traversing mechanism etc. on

	various data structures (Stack and Queue).
3	Illustrate efficient storage mechanisms of data for given problem like linked list.
4	Select appropriate sorting technique and searching technique for given problem
5	Illustrate concepts learned in various domains like DBMS, compiler construction etc.
6	Choose appropriate data structure for specified problem domain.

Subject code: **CSL304**	Subject: **Object Oriented Programming and Methodology Lab**	Credits:02

Course Objective:

At the end of course, student should be able to:

1	Describe Object oriented concepts like object, class, Inheritance, encapsulation, etc.
2	Explain fundamentals of object-oriented programming in Java, including defining classes, invoking methods, using class libraries, etc.
3	Use the Java JDK environment to create, debug and run simple Java programs.
4	Use various model of object oriented programming: abstract data types, encapsulation, inheritance and polymorphism
5	Categorize real-world scenarios using top down approach.
6	Design and develop GUI using different controls

Course Outcomes:

At the end of course, students will attain an ability to:

1	Use fundamental programming concept.
2	Illustrate the concept of packages, classes and objects.
3	Demonstrate the concept of inheritance and interfaces.
4	Explain the concept of string, array and vectors.
5	Implement the notation of exception handling and multithreading.
6	Develop GUI based application.

Class: TE

Subject code: **CSDLO5012**	Subject: **Advanced Operating systems**	Credits:04

Course Objective:

At the end of course, student should be able to:

1	Describe design issues of Advanced Operating systems.
2	Explain design aspects and data structures used for file subsystem, memory subsystem and process subsystem of Unix OS.
3	Explain different architectures used in Multiprocessor OS and analyze the design and data structures used in Multiprocessor operating systems.
4	Differentiate between threads and processes and compare different processor scheduling algorithms used in Multiprocessor OS
5	Classify and compare Real Time OS and analyze various real time scheduling algorithms.
6	Explore architectures and design issues of Mobile OS, Virtual OS, Cloud OS

Course Outcomes:

At the end of course, students will attain ability to:

1	State design issues of Advanced Operating systems.
2	Express design aspects and data structures used for file subsystem, memory subsystem and process subsystem of Unix OS
3	Explain different architectures used in Multiprocessor OS and analyze the design and data structures used in Multiprocessor operating systems.
4	Differentiate between threads and processes and compare different processor scheduling algorithms used in Multiprocessor OS
5	Classify and compare Real Time OS and analyze various real time scheduling algorithms.
6	Explore architectures and design issues of Mobile OS, Virtual OS, Cloud OS

Subjectcode:**CSC502**	Subject: **Database Management System**	Credits:04

Course Objective:

At the end of course, student should be able to:

1	Demonstrate the role of a database management system in an organization.
2	Design data modeling using the entity-relationship and developing database designs.
3	Demonstrate the use of Structured Query Language (SQL) and learn SQL syntax.
4	Construct simple and moderately advanced database queries using Structured Query Language (SQL).
5	Select normalization techniques to normalize the database.
6	Infer the needs of database processing and learn techniques for controlling the consequences of concurrent data access.

Course Outcomes:

At the end of course, students will attain an ability to:

1	Design and draw ER and EER diagram for the real life problem with software tool.
2	Construct and update database and tables with different DDL and DML statements.
3	Use integrity constraints and able to provide security to data.
4	Construct and execute Complex queries.
5	Design triggers and procedures for specific module/task
6	Integrate concurrent transactions and able to access data through front end (using JDBC ODBC connectivity.)

Subject code:**CSC501**	Subject : **Microprocessor**	Credits:04

Course Objective:

At the end of course, student should be able to:

1	Describe the basic concepts of microprocessor and assembly language
2	Know the instructions of basic microprocessor
3	Express the importance of different peripheral devices and their interfacing to 8086
4	Express multiprocessor and high end processor configurations
5	List techniques for faster execution of instructions and improve the speed of operation and performance of microprocessors
6	Outline the concept of multi-core processors.

Course Outcomes:

At the end of course, students will attain an ability to:

1	Know the overview of architecture and basic concepts of microprocessor

2	Implement programs to run on 8086 microprocessor based systems
3	Design systems using memory chip and peripheral chips
4	Understand and devise techniques for faster execution of instructions ,improve speed and enhance performance of microprocessor
5	Distinguish between RISC and CISC processors.
6	Describe multi-core processor and its advantages

Subject code:**CSC504**	Subject: **Theory of Computer Science**	Credits:04

Course Objective:

At the end of course, student should be able to:

1	Describe concepts of fundamentals of grammars and languages.
2	Develop concepts of theoretical design of deterministic and non-deterministic finite automata and Push down automata.
3	Discriminate between different types of grammars.
4	Develop understanding of different types of Turing machines and applications.
5	Understand the concept of Un decidability.
6	Design different types of computing machines.

Course Outcomes:

At the end of course, students will attain an ability to:

1	Identify the central concepts in theory of computation and differentiate between determini nondeterministic automata, also obtain equivalence of NFA and DFA.
2	Infer the equivalence of languages described by finite automata and regular expressions.
3	Devise regular, context free grammars while recognizing the strings and tokens.
4	Design pushdown automata to recognize the language.
5	Develop an understanding of computation through Turing Machine.
6	Understanding of decidability and un decidability.

Class: BE

Subject code:**CPC703**	Subject**: Artificial Intelligence**	Credits:04

Course Objective:

At the end of course, student should be able to:

1	Construct the basic ideas and techniques underlying the design of intelligent systems.
2	Make students understand and Explore the mechanism of mind that enable intelligent thought and action.
3	Know advanced representation formalism.
4	Express search techniques.
5	Design different models of reasoning.
6	Describe the methods to deal with uncertain and incomplete

Course Outcomes:

At the end of course, students will attain an ability to:

1	Develop a basic understanding of AI building blocks presented in intelligent agents.
2	Choose an appropriate problem-solving method.
3	Identify knowledge representation technique for specific condition.
4	Categorize the strength and weaknesses of AI approaches to knowledge– intensive problem solving.
5	Design models for reasoning with uncertainty as well as the use of unreliable information.
6	Implement and develop the AI applications in real world scenario.

Subject code:**CPC702**	Subject :**Cryptography & System Security**	Credits:04

Course Objective:

At the end of course, student should be able to:

1	Summarize the students with contemporary knowledge in Cryptography and Security.

2	Produce the basics of cryptography to keep networks, systems, and data secure.
3	Illustrate skills to design security protocols for recognize security problems.
4	Determine the general ideas behind Cryptographic hash function.
5	Describe how crypto can be used as an effective tool in providing assurance concerning privacy and integrity of information.
6	Revise security mechanisms & services related to security goals.

Course Outcomes:

At the end of course, students will attain an ability to:

1	Describe the principles and practices of cryptographic techniques.
2	Classify/Illustrate a variety of generic security threats and vulnerabilities and identify & analyze particular security problems for given application.
3	Use/Apply the application of security techniques and technologies in solving real- life security problems in practical systems.
4	Identify and Use appropriate security techniques to solve security problem.
5	Design security protocols and methods to solve the specific security problems.
6	Demonstrate with current research issues and directions of security.

Subject code:**CPC703**	Subject**: Digital Signal Processing.**	Credits:04

Course Objective:

At the end of course, student should be able to:

1	Understand the fundamental concepts of Digital Signal Processing.
2	Explain the properties of DFT in mathematical problem solving.
3	Use mathematical operation such as Linear convolution, Circular convolution, Linear convolution using circular convolution.
4	Illustrate FFT calculations mathematically and develop FFT based DSP algorithms.
5	Interpret DSP processor for real time signal processing application.
6	Explain real time signal processing.

Course Outcomes:

At the end of course, students will attain an ability to:

1	Summaries the concept of DT Signal and perform signal manipulation 4.
2	Compute analysis of DT system in time domain
3	Demonstrate mathematical operation such as Linear convolution, Circular convolution, Linear convolution using circular convolution.
4	Develop FFT flow-graph.

5	Develop Fast DSP Algorithms.
6	Design DSP system for Real Time Signal Processing.

Subject code:**CPC705**	Subject: **Soft Computing**	Credits:04

Course Objective:

At the end of course, student should be able to:

1	Conceptualize the working of human brain using ANN.
2	Demonstrate neural networks that can learn from available examples and generalize to form appropriate rules for inference systems.
3	Introduce the ideas of fuzzy sets, fuzzy logic and use of heuristics based on human experience.
4	Interpret the mathematical background for carrying out the optimization
5	Identify genetic algorithm for seeking global optimum in self-learning situation.
6	Learn applications of genetic algorithm

Course Outcomes:

At the end of course, students will attain an ability to:

1	Identify and appreciate the applications which can use fuzzy logic, Design inference systems
2	Demonstrate the difference between learning and programming, Explore practical applications of neural networks (NN).
3	Appreciate the importance of optimizations & its use in computer engineering fields and other domains
4	Demonstrate the efficiency of a hybrid system
5	Identify the neural network and fuzzy logic can be hybridized to form a neuro-fuzzy network
6	Design various applications GA.

SARASWATI Education Society's

SARASWATI College of Engineering

Learn Live Achieve and Contribute

Kharghar, Navi Mumbai - 410 210.

Course Objectives and Course Outcomes

Even Semester

Class: SE

Subject code:**CSC402**	Subject: **Analysis of Algorithms**	Credits:04

Course Objective:

At the end of course, student should be able to:

1	Explain students to the general tools and techniques for analyzing computer algorithms.
2	Illustrate the students with mathematical preliminaries required to analyze and design computer algorithms.
3	Choose the advanced data structures required to design efficient computer algorithms.
4	Summarize the students with specific algorithms for a number of important computational problems like sorting, searching, and graphs, … etc.
5	Describe the importance of designing efficient algorithms by comparing different complexity classes.
6	Categorize strategies for solving problems not solvable in polynomial time.

Course Outcomes:

At the end of course, students will attain an ability to:

1	Illustrate the running time and space complexity of algorithms.
2	Describe the complexity of divide and conquer strategy.
3	Identify the complexity of greedy strategy.
4	Determine the complexity of dynamic programming strategy.
5	Explain and apply backtracking, branch and bound and string-matching techniques to deal with some hard problems.
6	Categorize the classes P, NP, and NP-Complete and be able to prove that a certain Problem is NP-Complete.

Subject code: **CSC404**	Subject: **Computer Graphics**	Credits:04

Course Objective:

At the end of course, student should be able to:

1	Make students understand the fundamental knowledge and basic technical competence in the field of computer graphics.
2	Emphasize on implementation aspect of Computer Graphics Algorithms.
3	Prepare the student for advance areas like Image Processing or Computer Vision.
4	Prepare the student for Virtual Reality and professional avenues in the field of Computer Graphics.
5	Discuss 2D-3D dimensional geometric transformation.
6	Classify the interdisciplinary nature of computer graphics is emphasized in the wide variety of examples and applications.

Course Outcomes:

At the end of course, students will attain an ability to:

1	Illustrate the basic concepts of Computer Graphics.
2	Demonstrate various algorithms for scan conversion and filling of basic objects and their comparative analysis.
3	Invent knowledge about two and three dimensional transformations.
4	Choose geometric transformations, viewing and clipping on graphical objects.
5	Express solid model representation techniques and projections.
6	Illustrate visible surface detection techniques and illumination models.

Subject code:**CSC403**	Subject**: Computer Organization and Architecture**	Credits:04

Course Objective:

At the end of course, student should be able to:

1	Demonstrate the basic structure and operation of a digital computer.
2	Discus iscuss in detail the operation of the arithmetic unit including the algorithms & implementation of fixed-point and floating-point addition, subtraction, multiplication & division.
3	Illustrate the different ways of communicating with I/O devices and standard I/O interfaces.
4	Explain performance issues in processor and memory design of a digital computer.
5	Classify the hierarchical memory system including cache memories and virtual memory.
6	Infer processor performance improvement using instruction level parallelism

Course Outcomes:

At the end of course, students will attain an ability to:

1	Describe basic structure of the computer system.
2	Demonstrate the arithmetic algorithms for solving ALU operations.
3	Describe instruction level parallelism and hazards in typical processor pipelines.
4	Describe superscalar architectures, multi-core architecture and their advantages
5	Demonstrate the memory mapping techniques.
6	Identify various types of buses, interrupts and I/O operations in a computer system.

Subject code:**CSC405**	Subject**: Operating System**	Credits:04

Course Objective:

At the end of course, student should be able to:

1	Introduce basic concepts and functions of operating systems.
2	Classify the concept of process, thread and resource management.
3	Discus the concepts of process synchronization and deadlock.
4	Illustrate various Memory and File management techniques.
5	Explain various IO management techniques.
6	Differentiate different operating systems.

Course Outcomes:

At the end of course, students will attain an ability to:

1	Understand role of Operating System in terms of process, memory, file and I/O manageme
2	Explain and analyze the concept of a process, thread, mutual exclusion and deadlock.
3	Evaluate performance of process scheduling algorithms and IPC.
4	Identify and analyze the concepts of memory management techniques.
5	Determine the performance of memory allocation and replacement techniques.
6	Implement and analyze different techniques of file and I/O management.

SARASWATI Education Society's

SARASWATI College of Engineering

Learn Live Achieve and Contribute

Kharghar, Navi Mumbai - 410 210.

Class: TE

Subject code:**CPC603**	Subject**: Distributed Databases**	Credits:04

Course Objective:

At the end of course, student should be able to:

1	Enhance the previous knowledge of database systems by deepening the understanding of the theoretical and practical aspects of the database technologies.
2	Identify the need for distributed database technology to tackle deficiencies of the centralized database systems.
3	Understand basic principles and implementation techniques of distributed database systems.
4	Identify architecture, design issues and integrity control of distributed database.
5	Understand query processing and optimization, transactions, and concurrency control.
6	Understand basics of XML and how it can be used for data integration. Build simple XML.

Course Outcomes:

At the end of course, students will attain an ability to:

1	Analyze currently available models, technologies and approaches to build distributed database systems and services. Differentiate the different database systems and integrate them.
2	Construct simple and moderately advanced database queries using Structured Query Language (SQL).
3	Build XML for schema integration.
4	Provides solution for heterogeneous database
5	Apply learned skills to solving practical database related tasks
6	Design and implement distributed database for enterprise application.

Subject code:**CSC601**	Subject: **Software Engineering**	Credits:04

Course Objective:

At the end of course, student should be able to:

1	Give the knowledge of software engineering discipline.
2	Apply analysis, design and testing principles to software project development.
3	Demonstrate and evaluate real time projects with respect to software engineering principles.
4	Identify requirements and apply process model to selected case study.
5	Analyze and design models for the selected case study using UML modeling.
6	Use various software engineering tools.

Course Outcomes:

At the end of course, students will attain an ability to:

1	Understand and demonstrate basic knowledge in software engineering.
2	Identify requirements, analyze and prepare models.
3	Plan, schedule and track the progress of the projects.
4	Design & develop the software projects.
5	Identify risks, manage the change to assure quality in software projects.
6	Apply testing principles on software project and understand the maintenance concepts.

Subject code:**CPC601**	Subject: **System Programming and Compiler Construction**	Credits:04

Course Objective:

At the end of course, student should be able to:

1	Demonstrate the role and functioning of various system programs over application program.
2	Explain the functioning of assembler, macroprocessor.
3	Illustrate to initiate an understanding of compilers in general and brief about phases of compiler.
4	Determine the theoretical framework for optimizing the code.
5	Describe different Intermediate Code Generation techniques.
6	Familiarize and encourage the students to use various software tools for Developing System programs.

Course Outcomes:

At the end of course, students will attain an ability to:

1	Describe different system software.
2	Design assembler and macro-processor.
3	Use Lex tool for generating lexical analyzer and design new language structures with the help of grammars
4	Apply optimization principles on given code.
5	Identify role of Intermediate Code Generation in connection with language designing
6	Recognize various parser types and use YACC.

Class: BE

Subject code:**CPC801**	Subject :**Data warehousing & Mining**	Credits:04

Course Objective:

At the end of course, student should be able to:

1	Identify the methodology of engineering legacy databases for data warehousing.
2	Generalize data mining to derive business rules for decision support systems
3	Identify the data, identify the problems, and choose the relevant models and algorithms to apply
4	Describe business intelligence for an enterprise
5	Generalize web enabled data warehouse.
6	Use the research in data warehousing and data mining

Course Outcomes:

At the end of course, students will attain an ability to:

1	Generalize and implement classical algorithms in data mining and data warehousing
2	Generalize the strengths and weaknesses of the algorithms
3	Identify the application area of algorithms, and apply them
4	Learn data mining techniques as well as methods in integrating.
5	Interpreting the data sets and improving effectiveness, efficiency and quality for data
6	Design application for social and technical task.

Subject code:**CPC802**	Subject : **Human Machine Interaction**	Credits:04

Course Objective:

At the end of course, student should be able to:

1	Choose the good interface design.

2	Explain the importance of human psychology in designing good interfaces
3	Use HMI in their day – to – day activities
4	Invent creativity in each student – build innovative applications that are user friendly.
5	Interrelate research in Machine Interface Design
6	Classify about input and output devices .

Course Outcomes:

At the end of course, students will attain an ability to:

1	Design user centric interfaces.
2	Develop innovative and user friendly interfaces.
3	Implement HMI in their day-to-day activities
4	Define existing interface designs, and improve them.
5	Illustrate application for social and technical task.
6	Distinguish input and output devices .

Subject code: **CPE8031**	Subject : **Machine Learning**	Credits:04

Course Objective:

At the end of course, student should be able to:

1	Express the basic concepts and techniques of Machine Learning.
2	Classify with regression methods.
3	Identify different classification methods
4	Implement different clustering methods
5	Demonstrate Dimensionality reduction Techniques
6	Use concept of SVM

Course Outcomes:

At the end of course, students will attain an ability to:

1	Gain knowledge about basic concepts of Machine Learning
2	Identify machine learning techniques suitable for a given problem
3	Solve the problems using various machine learning techniques
4	Categorise Dimensionality reduction techniques.
5	Design application using machine learning techniques
6	Implement SVM .

Subject code: **CPC803**	Subject: **Parallel and Distributed Systems**	Credits:04

Course Objective:

At the end of course, student should be able to:

1	Explain students with contemporary knowledge in parallel and distributed systems
2	Make students understand with skills to analyze and design parallel and distributed applications.
3	Express master skills to measure the performance of parallel and distributed algorithms
4	Develop and apply knowledge of parallel and distributed computing techniques and methodologies.
5	Know performance measures for parallel systems.
6	Classify taxonomies of parallel systems.

Course Outcomes:

At the end of course, students will attain an ability to:

1	Categories the principles and concept in analyzing and designing the parallel and distributed system.
2	Make about ways to parallelize problems.
3	Gain an appreciation on the challenges and opportunities faced by parallel and distributed systems.
4	Implement the middleware technologies that support distributed applications such as RPC, RMI and object based middleware.
5	Improve the performance and reliability of distributed and parallel programs.
6	Use the application of fundamental Computer Science methods and algorithms in the development of parallel applications.