Description of all courses of the program

Programming Language I

A. Course General Information:

Course Code:	CSE110
Course Title:	Programming Language I Programming Language I Laboratory
Credit Hours (Theory + Lab):	3 + 0
Contact Hours (Theory + Lab):	3 + 3
Category:	Program Core
Type:	Required, Engineering, Lecture + Laboratory
Prerequisites:	N/A
Co-requisites:	None

B. Course Catalog Description (Content):

This course provides an introduction to the foundations of computation and purpose of mechanized computation. Emphasis will be placed on techniques of problem analysis and the development of algorithms and programs. Topics will include:
1. Introduction to digital computers and programming algorithms
2. Information representation in digital computers. Writing, debugging and running programs (including file handling) on various digital computers using an appropriate language.
3. Data structures (String, List, Tuple and Dictionary), iteration, as well as the design and analysis of the basics of algorithms.

The course includes a compulsory 3-hour laboratory work each week. Students will be expected to do homework assignments in problem solving and program design as well as weekly laboratory assignments to reinforce the lecture material.

C. Course Objective:

The objectives of this course are to

a. Introduction to underlying concepts of problem-solving tools, specially flow control design tools, its components, sequential programs, preliminary non-sequential algorithms.

b. Instrumentation of non-sequential program statements using flow control tools. Overview of information representation using data types and their manipulation using operators for program design.
c. Perception of syntax, tools and techniques used in procedural programming language.
d. Understanding of String, List, Tuple and Dictionary data structures, iterating over different data structures.
e. Becoming familiar with real life objects using objects, classes and their methodical behaviors.
f. Understanding of decomposition, abstraction and better modularization of huge programs using functions.
g. Understanding the basic concepts of exception/error handling.
h. Understanding the basic concepts of reading and writing files (File I/O).
i. Review of writing programs.

D. Course Outcomes (COs):

Upon successful completion of this course, students will be able to

CO 1	**Recognize** the concepts of Function designing and assemble various functions for solving a problem.
CO 2	**Analyze the** basics of exception/error handling and File I/O.
CO 3	**Identify** Data types, Operators.
CO 4	**Illustrate** the fundamental concepts of Loops.
CO 5	**Comprehension** of syntax, tools of procedural programming and the basics of searching and sorting algorithms.
CO 6	**Demonstrate** the fundamental concepts of String, List, Tuple and Dictionary data structure.

E. Mapping of CO-PO-Taxonomy Domain & Level- Delivery-Assessment Tool:

Sl.	CO Description	POs	Bloom's taxonomy domain/level	Delivery methods and activities	Assessment tools
CO1	**Recognize** the concepts of Function designing and assemble various functions for solving a problem.	PO2	Cognitive/ Create/ Psychomotor	Lectures, Discussion, Q/A, Lab tasks	Class work, Assessment, Assignment, Lab work
CO2	**Analyze the** basics of exception/error handling and File I/O.	PO2	Cognitive/ Understand	Lectures, Discussion, Q/A, Lab Tasks.	Class work, Lab work

CO3	**Identify** Data types, Operators.	PO1	Cognitive/ Understand	Lectures, Discussion, Lab Tasks	Class work, Lab Work
CO 4	**Illustrate** the fundamental concepts of Loops.	PO3	Cognitive/ Apply, Psychomotor/ Manipulation	Lectures, Discussion, Lab Tasks.	Assignment, Assessment, Lab Work
CO 5	**Comprehension** of syntax, tools of procedural programming and the basics of searching and sorting algorithms.	PO5	Cognitive/ Understand/ Applying	Lectures, Discussion, Lab Tasks.	Class work, Assignment, Lab Work
CO 6	**Demonstrate** the fundamental concepts of String, List, Tuple and Dictionary data structure.	PO2	Applying/ Understanding/ Psychomotor	Lectures, Discussion, Q/A, Lab Tasks	Class work, Assessment, Assignment, Lab Work

F. Course Materials:

i. Text and Reference Books:

Sl.	Title	Author(s)	Publication Year	Edition	Publisher	ISBN
1	Think Python	Allen B. Downey	2015	2nd ed.	O'Reilly Media, Inc.	9781491939369
2	Introduction to Computation and Programming Using Python with Application to Understanding Data	John V. Guttag	2017	2nd ed.	The MIT Press	9780262529624
3	Python Workbook A Brief Introduction with Exercises and Solutions	Ben Stephenson	2019	2nd ed.	Springer	9783030188726

ii. Other materials (if any)

a. Lecture notes and presentation slides
b. Lab sheets
c. Lab usage manual
d. IDE: Jupyter Notebook, Google Colab

G. Lesson Plan:

Course Outlines			Course Assessment Methods
No	**Topic details**	**Week**	Related to CO (if any)
1	Identify Data types, Operators. Introduction to underlying concepts of problem-solving tools. Instrumentation of non-sequential program statements. Perception of syntax, tools and techniques used in procedural programming language.	Week 1,2	CO3
2	Illustrate the fundamental concepts of Loops. Analyze loops, nested loops combined with branching to obtain program debugging skills.	Week 3	CO4
3	Demonstrate the fundamental concepts of String, List, Tuple and Dictionary data structure. Understanding different data structure, iterating over preset	Week 4, 5, 6, 7	CO6
4.	Design Functions and assemble various functions for solving a problem.	Week 8, 9	CO1
5.	Comprehension of the basics of searching and sorting algorithms	Week 10	CO5
6.	Comprehension of the basics of exception/error handling and File I/O	Week 11	CO2
7.	Review of writing and debugging programs.	Week 12, 13	

H. Assessment Tools:

Assessment Tools	Weightage (%)
Participation in class	5 %
Quizzes/Class Tests/Assignments	20 %
Mid Term Examination	20 %
Lab including Lab Assignments	25 %
Final	30 %

Total	100 %

I. CO Assessment Plan:

Assessment Tools	Course Outcomes					
	CO 1	CO 2	CO 3	CO 4	CO 5	CO 6
Examinations	X			X		X
Class work	X	X	X		X	X
Laboratory Work	X	X	X	X	X	X
Assignment	X			X		X

Programming Language II

A. Course General Information:

Course Code:	CSE111
Course Title:	Programming Language II Programming Language II Laboratory
Credit Hours (Theory+Lab):	3 + 0
Contact Hours (Theory+Lab):	3 + 3
Category:	Program Core
Type:	Required, Engineering, Lecture + Laboratory
Prerequisites:	CSE110 Programming Language I

B. Course Catalog Description (Content):

This course would be an introduction to data structures, formal specification and syntax of Object Oriented Programming (OOP), elements of language theory and mathematical preliminaries. Other topics that would be covered are formal languages, structured programming concepts, survey of features of existing high-level languages. Students would design and write applications using an appropriate language. The course includes a compulsory 3-hour laboratory work each week.

C. Course Objective:

The objectives of this course are to

a. introduce the students to the overall concept of OOP Programming Language.
b. introduce the students to the fundamental concepts of collections.
c. teach about different aspects of Access Modifiers.
d. teach about the fundamental concept of code reusability using inheritance.
e. familiarize in class designing using various characteristics of Python.
f. teach error/exception handling.

D. Course Outcomes (COs):

Upon successful completion of this course, students will be able to

Sl.	CO Description	Weightage (%)
CO1	Illustrate fundamental concepts of collections.	**5%**
CO2	Differentiate various aspects and syntax of OOP.	**5%**
CO3	Identify different types of Access Modifiers.	**10%**
CO4	Design the fundamental concepts of basic OOP characteristics using Python.	**40%**
CO5	Illustrate class design and code reusability using inheritance.	**10%**

E. Mapping of CO-PO-Taxonomy Domain & Level- Delivery-Assessment Tool:

Sl.	CO Description	POs	Bloom's taxonomy domain/level	Delivery methods and activities	Assessment tools
CO1	Illustrate fundamental concepts of collections.	c	Cognitive/Analyze	Notes,Lab class	Quiz, Exam, Lab work
CO2	Differentiate various aspects and syntax of OOP (Object Oriented Programming)	d	Cognitive/Analyze	Lectures, notes, Lab class	Quiz, Exam, Lab work
CO3	Identify different types of Access Modifiers.	a	Cognitive/Understand	Lectures, notes, Lab class	Quiz, Exam, Lab work
CO4	Design the fundamental concepts of basic OOP characteristics using Python (inheritance, polymorphism, abstraction, encapsulation)	c	Cognitive/Create	Lectures, notes, Lab class	Quiz, Exam, Lab work
CO5	Illustrate class design and code reusability using inheritance.	c	Cognitive/Analyze	Lectures, notes, Lab class	Quiz, Exam, Lab work

F. Course Materials:

i. Text and Reference Books:

Sl.	Title	Author(s)	Publication Year	Edition	Publisher	ISBN
1	Python Crash Course A Hands-On, Project-Based Introduction to Programming	Eric Matthes	2019	2nd	No Starch Press	ISBN-13: 9781593279288
2	Introduction to Computation and Programming Using Python.	John V. Guttag	2016	2nd	The MIT Press	ISBN-13 : 978-0262529624

ii. Other materials (if any)

a. Lecture notes and presentation slides
b. Lab hand-outs
c. Lab usage manual
d. Programming IDE (Jupyter Notebook, Google Colab, Spider)

G. Lesson Plan:

No	Topic	Week/Lecture#	Related CO (if any)
	Introduction to problem solving	Week 1	
	Review of programming basics	Week 2	
	Objects and Classes	Week 3, 4	CO1
	Access Specification	Week 5	CO2
	Review	Week 6	
Midterm			
	Method Overloading, Inheritance	Week 7	CO3
	Polymorphism, Dynamic method dispatch	Week 8	CO3,CO4
	Practice	Week 9	

	Abstract Class	Week 10	CO3, CO4
	Exception handling	Week 11	CO3
	Review	Week 12	
Lab	Practice using Modern Tools	Lab classes	CO5
Final Exam			

H. Assessment Tools:

Assessment Tools	**Weightage (%)**
Class Performance & Attendance	5
Quizzes and Assignments	20
Midterm Exam	20
Final Exam	30
Lab including Lab Assignments	25

I. CO Assessment Plan:

Assessment Tools	**Course Outcomes**							
	CO1	**CO2**	**CO3**	**CO4**	**CO5**	**CO6**	**CO7**	**CO8**
Quizzes (Q)	√	√	√	√	√	√	√	
Homework	√	√			√	√		
Midterm exam	√	√	√	√				
Final Exam					√	√	√	√
Project work								
H/W Lab work								
S/W Lab work	√	√	√	√	√	√	√	√

Data Structures

A. Course General Information:

Course Code:	CSE 220
Course Title:	Data Structures
Credit Hours (Theory+Lab):	3 + 0
Contact Hours (Theory+Lab):	3 + 3
Category:	Program Core
Type:	Required, Engineering , Lecture + laboratory
Prerequisites:	CSE 111 Programming Language II

B. Course Catalog Description (Content):

This course is an introduction to data structures, where the students will study the elementary data structures such as arrays, lists, stacks, queues, trees, etc. These data structures will be used to study and implement different algorithms such as sorting, searching, tree traversal, etc. The course includes a 3 hour mandatory laboratory per week as CSE220L. In the laboratory, the students will use a standard programming language, usually Java, to implement the various data structures and algorithms learned in the theory component of the course.

C. Course Objective:

a. Teach students the basics of circular array and advantage(s) of it over a linear array

b. Demonstrate the construction and manipulation of different types of linked lists

c. Introduce the students to stack and queue data structures and explain how they are implemented

d. Discuss the fundamental concept of recursion so that they can build recursive models for simple problems.

e. Discuss trees and explain related algorithms

D. Course Outcomes (COs):

Upon successful completion of this course, students will be able to

Sl.	CO Description	Weightage (%)
CO1	**Show** different operations such as insertion, removal, rotation, shifting, etc. on linear arrays, circular arrays, linked lists, Stack, Queue and Tree data structures.	25%
CO2	**Demonstrate** basic algorithms related to searching and tree traversal using various data structures.	25%
CO3	**Compare** the suitability and merits of various data structures and basic algorithms when given certain requirements or constraints.	10%
CO4	**Apply** concepts of recursion to solve programming problems.	15%
CO5	**Construct** general-purpose data structures and basic algorithms for solving programing problems.	25%

E. Mapping of CO-PO-Taxonomy Domain & Level- Delivery-Assessment Tool:

Sl.	CO Description	POs	Bloom's taxonomy domain/level	Delivery methods and activities	Assessment tools
CO1	**Show** different operations such as insertion, removal, rotation, shifting, etc. on linear arrays, circular arrays, linked lists, Stack, Queue and Tree data structures.	**a**	Cognitive/ Apply	Lecture + Lab	Assignment, Quiz, Exam, Lab Work
CO2	**Demonstrate** basic algorithms related to searching, tree traversal using various data structures.	**a**	Cognitive/ Apply	Lecture + Lab	Assignment, Quiz, Exam, Lab Work
CO3	**Compare** the suitability and merits of various data structures and basic algorithms when given certain requirements or constraints.	**a**	Cognitive/Evaluate	Lecture	Exam
CO4	**Apply** concepts of recursion to solve programming problems.	**a**	Cognitive/ Apply	Lecture + Lab	Assignment, Quiz, Exam, Lab Work
CO5	**Construct** general-purpose data structures and basic algorithms for solving programing problems.	**c**	Cognitive/ Create	Lab	Lab work

F. Course Materials:

i. Text and Reference Books:

Sl.	Title	Author(s)	Publication Year	Edition	Publisher	ISBN
1	Algorithms in Java	Robert Sedgewick and Kevin Wayne	2011	4th Edition	Addison-Wesley	ISBN-10: 032157351X ISBN-13: 9780321573513
2	Introduction to Algorithms	Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein	2009	3rd Edition	MIT Press	ISBN-10: 0262033844 ISBN-13: 9780262033848

G. Lesson Plan (Theory):

No	Topic	Week/Lecture#	Related CO (if any)
1	Arrays and Circular Arrays	Lecture 1, 2, 3	CO1, CO3, CO5
2	Linked lists	Lecture 4, 5, 6, 7, 8	CO1 CO3, CO5
3	Stacks (using arrays and linked lists)	Lecture 9, 10	CO1 CO3, CO5
4	Queues (using arrays and linked lists)	Lecture 11, 12	CO1 CO3, CO5
Review and Midterm Exam			
5	Recursion	Lecture 13, 14	CO4, CO5
6	Hash table and Hashing	Lecture 15, 16	CO2 CO3, CO5
7	Introduction to Trees and BST	Lecture 17, 18	CO1, CO2, CO5
8	Introduction to Heap	Lecture 19, 20	CO1, CO2, CO5
Review and Final Exam			

Lesson Plan (Laboratory):

No	Topic	Week/Lecture#
1	Linear array, Circular array	Week 1
2	Singly Linked List (basic)	Week 2
3	Singly Linked List (basic) Continuation	Week 3
4	Dummy headed circular doubly linked list	Week 4
5	Build a stack using List Use the stack for parentheses checking	Week 5
6	Build a queue using a circular array. Apply the queue in a problem	Week 6
7	Lab Midterm Assessment	Week 7
8	Recursion	Week 8
9	Searching, Sorting	Week 9
10	Key-indexing, Hashing	Week 10
11	Tree, graph	Week 11
12	Lab Final Assessment	Week 12

H. Assessment Tools:

Assessment Tools	**Weightage (%)**
Class Participation & Attendance	5%
Quizzes	10-15%
Midterm Exam	20%
Assignment	5-10%
Lab	25%
Final Exam	30%

I. CO Assessment Plan:

Assessment Tools	**Course Outcomes**				
	CO1	**CO2**	**CO3**	**CO4**	**CO5**
Assignments	√	√		√	
Quizzes	√	√		√	
Midterm Exam	√	√			√
Lab	√	√		√	

Final Exam	√	√	√	√	

Algorithm Analysis & Design

A. Course General Information:

Course Code:	CSE 221
Course Title:	Algorithm Analysis & Design
Credit Hours (Theory+Lab):	3 + 0
Contact Hours (Theory+Lab):	3 + 3
Category:	Programe Core
Type:	Required, Engineering , Lecture + labs
Prerequisites:	CSE220

B. Course Catalog Description (Content):

This course addresses the study of efficient algorithms, their analyses and effective algorithm design techniques. Standard algorithm design strategies, such as, Divide and Conquer paradigm, Greedy method, Dynamic programming, Backtracking, Basic search and traversal techniques, Graph algorithms, Elementary parallel algorithms, Algebraic simplification and transformations, Lower bound theory, NP-hard and NP-complete problems are discussed in the course. Examples of data structures and algorithms studied in details are Heaps; Hashing; Graph algorithms: Shortest paths, Depth-first and Breadth-first search, Network flow, Computational geometry, Minimum Spanning Tree; Integer arithmetic: GCD, primality; polynomial and matrix calculations; Sorting; Performance bounds, asymptotic analysis, worst case and average case behavior, correctness and complexity. The course includes a compulsory 3 hour laboratory work every week.

C. Course Objective: The objectives of this course are to :

a. introduce students to time and space complexity of algorithms
b. teach students different sorting and searching methods and make them understand which is effective to use.
c. make them familiar with different problem solving paradigms as described in the course catalog above

D. Course Outcomes (COs):

Upon successful completion of this course, students will be able to

Sl.	CO Description	Weightage (%)
CO1	**Examine** the pseudocode of an algorithm to be able to understand how it works	10%
CO2	**Demonstrate** the level of understanding by simulating the algorithm in a given scenario	20%
CO3	**Analyze** the time complexity of an algorithm for comparing it with similar algorithms.	20%
CO4	**Apply** different problem solving approaches to solve real world problems involving the use of graphs, trees, or other advanced data structures	25%
CO5	**Convert** different algorithms into computer codes and successfully execute them	25%

E. Mapping of CO-PO-Taxonomy Domain & Level- Delivery-Assessment Tool:

Sl.	CO Description	POs	Bloom's taxonomy domain/level	Delivery methods and activities	Assessment tools
CO1	**Examine** the pseudocode of an algorithm to be able to understand how it works.	**a**	Cognitive / Apply	Lecture + Lab Works	Quiz , Exam
CO2	**Demonstrate** the level of understanding by simulating the algorithm in a given scenario	**a**	Cognitive / Apply	Lecture + Lab Works	Quiz , Exam
CO3	**Analyze** the time complexity of an algorithm for comparing it with similar algorithms.	**a**	Cognitive / Analyze	Lecture + Lab Works	Quiz , Exam

CO4	**Apply** different problem solving approaches to solve real world problems involving the use of graphs, trees, or other advanced data structures	**e**	Cognitive / Apply	Lecture + Lab Works	Quiz , Exam, Lab Tasks
CO5	**Convert** different algorithms into computer codes and successfully execute them	**a**	Cognitive / Apply	Lecture + Lab Works	Lab Tasks

F. Course Materials:

i. Text and Reference Books:

Sl.	Title	Author(s)	Publication Year	Edition	Publisher	ISBN
1	Introduction to Algorithms	Charles E. Leiserson, Clifford Stein, Ronald Rivest, and Thomas H. Cormen	2009	3rd edition	The MIT Press	ISBN: 9780262033848
2	Fundamentals of Computer Algorithms	Horowitz and Sahani	2008	2nd Edition	Universities Press	ISBN: 9780929306414
3	Algorithm Design	Eva Trados and John Klenberg	(March 26, 2005)	1 edition	Pearson;	ISBN-13: 978-0321 295354

ii. Other materials (if any)

a. **lecture note + slides**

b. **visual algo simulation site.**

G. Lesson Plan:

No	Topic	Week/Lecture	Related CO
1	Introduction & Algorithm Analysis; Time Complexity, Space Complexity Analysis; Recursion and Backtracking;	Week 0-1	CO1-5

2	Sorting and Searching: Bubble, Selection and Insertion Sorting; Linear and Binary Searching Algorithms and its variants with time complexity. Divide and Conquer Basics; Merge and Quick Sort with derivation of running time;	Week 2-3	CO1-5
3	Graph Basics: Types of Graphs, Data Structures used BFS , DFS, and applications: Edge classification, cycle detection, bipartite/bicolorable graph.	Week 4-5	CO1-5
Mid Exam			
4	DAG, Topological sort, Strongly Connected Components (Kosaraju, Tarjan) Shortest path Dijkstra, Negative cycle: Bellman-Ford	Week 7	CO1-5
4	Minimum spanning Tree using Kruskal's Algorithm; Disjoint Set Data Structure Minimum Spanning Tree using Prim's Algorithm;	Week 8	CO1-5
5	Introduction to greedy, time scheduling interval; Fractional knapsack, Huffman encoding decoding	Week 9	CO1-5
6	Dynamic Programming: Basics; Knapsack 0/1, LCS, Coin Change (how many ways, minimum no. of coins) Recursive and iterative DP formulation, comparison	Week 10-11	CO1-5
7	P vs NP	Week 12	CO3
Final Exam			

H. Assessment Tools:

Assessment Tools	**Weightage (%)**
Class Participation/Performance	5%
Assignment	5%

Quiz	15%
Midterm Exam	20%
Lab	25%
Final Exam	30%

I. CO Assessment Plan:

Assessment Tools	Course Outcomes				
	CO1	CO2	CO3	CO4	CO5
Class Participation/Performance					
Assignment					
Quiz					
Midterm Exam		√	√	√	
Lab				√	√
Final Exam		√	√	√	

Discrete Mathematics

A. Course General Information:

Course Code:	CSE 230
Course Title:	Discrete Mathematics
Credit Hours (Theory+Lab):	3 + 0
Contact Hours (Theory+Lab):	3 + 0
Category:	Program Core
Type:	Required, Engineering, Lecture
Prerequisites:	None
Co-requisites:	None

B. Course Catalog Description (Content):

Set theory, Elementary number theory, Graph theory, Paths and trees, Boolean Algebra, Binary Relations, Functions, Algebraic system, Generating functions, Induction, Reduction, Semigroup, Permutation groups, Discrete Probability, Mathematical logic, Prepositional calculus and Predicate calculus.

C. Course Objective:

The objective of the course is to:

a) Make students understand the mathematical reasoning about basic data types (such as numbers, sets, etc.)
b) Show the students how to synthesize elementary proofs using various proving techniques.
c) Teach them to model and analyze the computational processes using analytic and combinatorial methods.
d) Make the students apply principles of discrete probability to calculate probability and expectations of simple random processes.
e) Introduce the method of using and analyzing recursive definitions.

f) Show the application of discrete mathematics in design and analysis of algorithm, computability theory.

D. Course Outcomes (COs):

Upon successful completion of this course, students will be able to

Sl.	CO Description	Weightage (%)
CO1	Use logical notation to define and reason about those fundamental concepts	5
CO2	Analyze computational processes using elementary combinatorial methods	15
CO3	Evaluate probabilities and discrete distributions for simple processes and calculate expectations	20
CO4	Synthesize induction hypothesis and simple induction proofs	10
CO5	Evaluate elementary mathematical arguments and identify fallacious reasoning and conclusion	15
CO6	Formulate and solve recurrence relations	15
CO7	Prove elementary properties of Modular arithmetic and explain their application in computer science (cryptology and hashing algorithm)	15

E. Mapping of CO-PO-Taxonomy Domain & Level- Delivery-Assessment Tool:

Sl.	CO Description	POs	Bloom's taxonomy domain/level	Delivery methods and activities	Assessment tools
CO1	**Use** logical notation to define and reason about fundamental mathematical concepts(sets,relations,functions,structures)	PO1	Cognitive/ Apply	Lecture, Discussion	Assignment, classwork
CO2	**Analyze** computational processes using elementary combinatorial methods	PO2	Cognitive/ Evaluate	Lectures, Discussions, Q/A	Assignment, Quiz, Exam
CO3	**Evaluate** probabilities and discrete distributions for simple processes and calculate expectations	PO1	Cognitive/ Evaluate	Lectures, Discussions, Notes	Assignment, Quiz, Exam

CO4	**Synthesize** induction hypothesis and simple induction proofs	PO2	Cognitive/ Create	Lectures, Discussions, Q/A, Notes	Assignment, Quiz, Exam
CO4	**Evaluate** elementary mathematical arguments and identify fallacious reasoning and conclusion	PO2	Cognitive/ Evaluate	Lectures, Discussions	Classwork, Quiz, Exam
CO6	**Formulate** and solve recurrence relations	PO2	Cognitive/ Create	Lectures, Discussions, Q/A	Assignment, Quiz, Exam
CO7	**Prove** elementary properties of Modular arithmetic and explain their application in computer science (cryptology and hashing algorithm)	PO1	Cognitive/ Apply	Lectures, Discussions, Q/A	Assignment, Quiz, Exam

F. Course Materials:

i. Text and Reference Books:

Sl.	Title	Author(s)	Publication Year	Edition	Publisher	ISBN
1	Elementary Probability for Applications	Rick Durrett	2009	1st	Cambridge University Press	978-0521867566
2	Discrete Mathematics and its application	Kenneth H. Rosen	2011	7th	Mc-Graw Hill Education	978-0073383095

ii. Other materials (if any)

- Lectures
- Presentation Slides

G. Lesson Plan:

No	Topic	Week/ Lecture#	Related CO (if any)
1	Set theory and functions	Lecture # 1-2	CO1
2	Combinatorics (Counting, Combinations, etc)	Lecture # 3-4	CO2
3	Combinatorial and Conditional Probability	Lecture # 5-10	CO3
4	Induction (Weak and Strong)	Lecture # 11-12	CO4

	Mid-Term		
5	Propositional Logic	Lecture # 13-14	CO5
6	Recurrence (formation and solve)	Lecture # 15-18	CO6
7	Modular Arithmatic and Application	Lecture # 19-22	CO7
8	Pigeonhole Principle	Lecture # 23-24	CO5

(** Please insert separate table of lesson plan for lab component if there is any)

H. Assessment Tools:

No.	Assessment Tools	Weightage (%)
1	Participation in class	5
2	Class performance	5
3	Quizzes/ Class Tests	20
4	Assignments	20
5	Mid-Term	20
6	Final	30
	Total	100

I. CO Assessment Plan:

Assessment Tools	Course Outcomes					
	CO1	**CO2**	**CO3**	**CO4**	**CO5**	**CO6**	**CO7**
Class performance	✔				✔		
Quizzes		✔	✔	✔	✔	✔	✔
Assignments	✔	✔	✔	✔		✔	✔
Exams		✔	✔	✔	✔	✔	✔

Circuits and Electronics

A. Course General Information:

Course Code:	CSE250
Course Title:	Circuits and Electronics Circuits and Electronics Laboratory
Credit Hours (Theory+Lab):	3 + 0
Contact Hours (Theory+Lab):	3 + 3
Category:	Program Core
Type:	Required, Engineering, Lecture + Laboratory
Prerequisites:	PHY112: Principles of Physics II
Co-requisites:	None

B. Course Catalog Description (Content):

Fundamental electrical concepts and measuring units. Direct current: voltage, current, resistance and power. Laws of electrical circuits and methods of network analysis; Introduction to magnetic circuits. Alternating current: instantaneous and R.M.S. current, voltage and power, average power for various combinations of R, L and C circuits, phasor representation of sinusoidal quantities. The course includes a compulsory 3 hour laboratory work alternate week.

C. Course Objective:

The objectives of this course are to:

a. To describe basic linear electrical circuit components such as, dependent and independent voltage and current sources, resistors, capacitors and inductors. I-V characteristics of these elements.

b. To familiarize students with basic electrical parameters such as voltage, current and power and passive sign convention for computing these parameters.

c. To explain basic laws like Ohm's law, Kirchhoff's voltage and current law, Thevenin's and Norton's theorem, Maximum power transfer theorem, Superposition theorem as well as voltage/current divider rule and characteristic equations for capacitors/inductors.

d. To introduce several circuit solving methods such as Source transformation, Nodal and Mesh analysis, method using Superposition theorem, Thevenin/Norton's equivalent circuit that takes advantage of basic laws.

e. To teach how to apply circuit solving techniques to solve steady-state circuits which contain resistors and dependent sources with direct current.

f. To analyze first order transient circuits with resistors, capacitors and inductors in the time domain.

g. To introduce phasors and analyzing circuits containing resistors, capacitors and inductors in phasor domain with alternating current.

D. Course Outcomes (COs):

Upon successful completion of this course, students will be able to

Sl.	CO Description	Weightage (%)
CO1	**Describe** the behavior of basic linear electrical circuit components.	10
CO2	**Identify** basic circuit parameters like voltage, current and power.	20
CO3	**Explain** basic laws of electricity which are used in circuit problems	20
CO4	**Apply** circuit solving techniques that use electrical laws to **evaluate** the circuit parameters of a given circuit.	25
CO5	**Analyze** the behavior of a first order transient circuit.	10
CO6	**Evaluate** circuit parameters in phasor domain in an AC circuit.	10
CO7	**Perform** in a team to conduct electrical circuit experiments	5

E. Mapping of CO-PO-Taxonomy Domain & Level- Delivery-Assessment Tool:

Sl.	CO Description	POs	Bloom's taxonomy domain/level	Delivery methods and activities	Assess ment tools
CO1	**Describe** the behavior of basic linear electrical circuit components.	**PO1**	Cognitive / Knowledge	Lecture/Discussion /Lab Task	Quiz
CO2	**Identify** basic circuit parameters like voltage, current and power.	**PO1**	Cognitive / Knowledge	Lecture/Discussion	Quiz
CO3	**Explain** basic laws of electricity which are used in circuit problems	**PO2**	Cognitive / Knowledge	Lecture/Discussion /Lab Task	Quiz, Lab Work
CO4	**Apply** circuit solving techniques that uses electrical laws to evaluate the circuit parameters of a given circuit.	**PO3**	Cognitive / Apply, Cognitive / Evaluate	Lecture/Problem Solving/Lab Task	Quiz, Exam

CO5	**Analyze** the behavior of a first order transient circuit.	**PO2**	Cognitive / Analyze	Lecture/Problem Solving/Lab Task	Quiz, Exam, Lab Work
CO6	**Evaluate** circuit parameters in phasor domain in an AC circuit.	**PO3**	Cognitive / Evaluate	Lecture/Problem Solving	Quiz, Exam
CO7	**Perform** in a team to conduct electrical circuit experiments	**PO9**	Affective/Responding	Lab class	Peer-review

F. Course Materials:

i. Text and Reference Books:

Sl.	Title	Author(s)	Publication Year	Edition	Publisher	ISBN
1	Fundamentals of Electric Circuits	Charles K. Alexander, Matthew N. O. Sadiku	2019	6th	McGraw Hill Education	978-9353165505
2	Introductory Circuit Analysis	Robert L. Boylestad	2013	12th	Pearson Education India	978-9332518612
3	Foundations of Analog and Digital Electronic Circuits	Anant Agarwal, Jeffrey H. Lang	2005	1st	Morgan Kaufmann Publishers	978-1558607354

ii. Other materials (if any)

a. Lecture notes
b. Lab hand-outs
c. Everycircuit (simulation software)
d. Tinkercad (simulation software)
e. LTSPICE (simulation software)

G. Lesson Plan:

No	Topic	Week/Lecture#	Related CO (if any)
1	Illustrating the motivation behind taking this course. What are the real-life implications of this course materials?	Week 1/Lecture 1	
2	Discuss basic circuit parameters like voltage, current, energy and power definitions and units. Introducing passive sign convention, positive-negative voltage/current/power. Discuss different types of circuit elements (active, passive), different types of	Week 1/Lecture 2	CO1, CO2

	sources (DC/AC, voltage/current, dependent/independent). Introducing circuit symbols.		
3	Introducing basic electrical components: resistors, voltage Source, current Source. I-V characteristics of a circuit element. Basic laws of electrical circuits: Ohm's law. Using Ohm's law to find power. Discuss various circuit configurations: Series, Parallel, Δ-Y etc. How to identify them and calculate equivalent resistance. Open and short circuit	Week 2/Lecture 1	CO2, CO3
4	Defining Node/Supernode. Introducing Current Sign Convention. Basic laws of electrical circuits: Kirchhoff's current law. Statement and application of KCL. Current divider rule in parallel circuit. Illustrating convention doesn't change the KCL equation. Usefulness of supernode.	Week 2/Lecture 1	CO3, CO4
5	Defining Mesh/Supermesh. Revisiting Passive Sign Convention. Basic laws of electrical circuits: Kirchhoff's voltage law. Statement and application of KVL. Voltage divider rule in series circuit. Illustrating the assumption of current direction doesn't change the KVL equation. Usefulness of supermesh.	Week 3/Lecture 1	CO3, CO4
Quiz 1			
6	Open circuit, Short circuit I-V characteristics. Idea of circuit equivalence. Equivalence with inactive current/voltage sources. Series-parallel equivalent circuit for resistance/voltage source/current source. Ideal/ non-Ideal current/voltage source. Calculating equivalent resistance of series-parallel circuit. Basic circuit theorem: Source Transformation theorem. Failure of applying in Wheatstone bridge circuit.	Week4/Lecture 1	CO1, CO2
7	Explaining Nodal Analysis technique, using it to solve for current, voltage, power in a given circuit (multiple examples)	Week 4/Lecture 2	CO2, CO4
8	Reintroducing dependent sources. Demonstrating Nodal Analysis with dependent sources. Problems with floating voltage sources, using Supernodes to solve such circuits.	Week 5/Lecture 1	CO2, CO4
9	Explaining MeshAnalysis technique, using it to solve for current, voltage, power in a given circuit (multiple examples).	Week 5/Lecture 2	CO2, CO4
10	Demonstrating Mesh Analysis with dependent sources. Problems with common current sources, using Supermeshes to solve such circuits.	Week 6/Lecture 1	CO2, CO4
Quiz 2			
Midterm			

11	Linear circuit elements. I-V characteristics of linear circuits. Circuit Theorems: Thevenin's theorem. Motivation behind Thevenin's theorem.	Week 7/Lecture 1	CO3, CO4
12	Using Thevenin's theorem for solving circuits. Condition for maximum power transfer. Norton's theorem, relation between Thevenin's and Norton's theorem.	Week 7/Lecture 2	CO2, CO4
13	Using test voltage/current sources while applying Thevenin's and Norton's theorem. Solving resistance matching problems for maximum power transferring.	Week 8/Lecture 1	CO4
14	Reintroduction to circuit linearity, linearity of voltage, current in circuits, non-linearity of power. Circuit theorem: Superposition theorem. Using superposition theorem for solving DC circuits.	Week 8/Lecture 2	CO3, CO4
15	Solving circuits using superposition theorem with dependent sources.	Week 9/Lecture 1	CO4
Quiz 3			
16	Capacitors and Inductors, their component equations. SI unit for measuring capacitance and inductance. Transient circuits, visualizing and analyzing transient circuits.	Week 10/Lecture 1	CO5
17	Response of transient circuit: first order (RC/RL) circuit, time constant. Analyzing and POtting first order transient circuit response.	Week 10/Lecture 2	CO5
18	Complex number review. Alternating current, importance of AC circuit. Visualizing the dynamics of an AC circuit.	Week 11/Lecture 1	CO1, CO2
19	Phasor diagram, introducing Impedance. Defining impedance for different elements Phasor analysis of an AC circuit. Instantaneous voltage, current and power.	Week 11/Lecture 2	CO6
20	Peak RMS voltage/current, real and reactive power. Applying superposition theorem on AC circuits containing sources of different frequencies.	Week 12/Lecture 1	CO2, CO6
Quiz 4			
Final Exam			

H. Assessment Tools:

Assessment Tools	Weightage (%)
Participation in class	10
Quiz	10
Assignment	5
Midterm Examination	25
Lab Work	20
Final Examination	30
	Total = 100%

I. CO Assessment Plan:

Assessment Tools	Course Outcomes					
	CO1	CO2	CO3	CO4	CO5	CO6
Quiz	✔	✔	✔	✔	✔	
Assignment			✔			✔
Midterm Examination	✔	✔	✔	✔		
Lab Work	✔		✔		✔	
Final Examination	✔	✔	✔	✔	✔	✔

Electronic Devices and Circuits

A. Course General Information:

Course Code:	CSE251 CSE251L
Course Title:	Electronic Devices and Circuits
Credit Hours (Theory+Lab):	3 + 0
Contact Hours (Theory+Lab):	3 + 3
Category:	Program Core
Type:	Required, Engineering, Lecture + Laboratory
Prerequisites:	CSE250
Co-requisites:	None

B. Course Catalog Description (Content):

Introduction to semiconductors, p-type and n-type semiconductors; p-n junction diode characteristics; Diode applications: half and full wave rectifiers, clipping and clamping circuits, regulated power supply using zener diode, diode-logic circuits. Bipolar Junction Transistor (BJT): principle of operation, I-V characteristics; Transistor circuit configurations (CE, CB, CC), BJT biasing, load lines; BJTs at low frequencies; Hybrid model, h parameters, simplified hybrid model; Small-signal analysis of single and multi-stage amplifiers, frequency response of BJT amplifier. Field Effect Transistors (FET): principle of operation of JFET and MOSFET; Depletion and enhancement type NMOS and PMOS; biasing of FETs; Low and high frequency models of FETs, Switching circuits using FETs; Operational Amplifiers (OPAMP): linear applications of OPAMPs – summer, subtractor, differentiator, integrator; gain, input and output impedances, active filters, frequency response and noise. The course includes a compulsory 3 hour laboratory work each week.

C. Course Objective:

The objectives of this course are to:

a. Introduce Electronic Devices such as Diodes and Transistors, and, semiconductor physics principles used to make them
b. Introduce the Piece-Wise Linear modeling technique to analyze circuits with non-linear devices
c. Show the application of diodes in constructing various circuits, such as, rectifiers, regulators, clippers, clampers, etc.
d. Show the application of transistors in building switching circuits and amplifiers with appropriate biasing methods.
e. Introduce students to the Operational Amplifier, and, using it in different circuits to perform analog signal-processing tasks, such as, Summing, Subtracting, Exponentiating, etc.
f. Training students to prototype circuits in hardware and analyzing their behavior
g. Exposing students to Circuit simulation tools to aid them in analyzing circuit behavior before implementing them in real life.

D. Course Outcomes (COs):

Upon successful completion of this course, students will be able to

Sl.	**CO Description**	**Weightage (%)**
CO1	Analyze the behavior of circuits with non-linear electronic devices using piece-wise linear models of such devices	25%
CO2	Design passive diode circuits for power-generation and analog signal-processing applications – rectifiers, regulators, clippers and clampers.	15%
CO3	Understand the two principal uses of transistors – switching and amplification, and, differentiate between the major transistor types – BJTs and MOSFETs.	10%
CO4	Design simple single-stage and multi-stage amplifier circuits (with appropriate biasing) using BJTs and MOSFETs	10%
CO5	Design Op-Amp Circuits to perform arithmetic operations on Analog Signals, e.g, Summing, Subtracting, Exponentiation, etc.	20%
CO6	Prototype electronic circuits on breadboards using basic electrical equipment (such as DC-Supply, Function-Generators, etc.) and, match their behavior with theoretical models	15%
CO7	Use circuit simulation tools to design complicated circuits and perform different types of analyses (DC, AC, Transient, etc.) to predict their behavior before implementing them in real life	5%

E. Mapping of CO-PO-Taxonomy Domain & Level- Delivery-Assessment Tool:

Sl.	CO Description	POs	Bloom's taxonomy domain/level	Delivery methods and activities	Assessment tools
CO1	Analyze the behavior of circuits with non-linear electronic devices using piece-wise linear models of such devices	PO2	Cognitive/Analyze, Apply	Lectures, Notes/Handouts, Simulation Demo	Quiz, Exam, Assignment
CO2	Design passive diode circuits for power-generation and analog signal-processing applications – rectifiers, regulators, clippers and clampers.	PO3	Cognitive/Create,	Lectures, Notes/Handouts, Simulation Demo,	Quiz, Exam, Assignment
CO3	Understand the two principal uses of transistors – switching and amplification, and differentiate between the major transistor types – BJTs and MOSFETs.	PO1	Cognitive/Evaluate, Apply	Lectures, Notes/Handouts, Simulation Demo	Quiz, Q/A, Exam
CO4	Design simple single-stage and multi-stage amplifier circuits (with appropriate biasing) using BJTs and MOSFETs	PO3	Cognitive/Create,	Lectures, Notes/Handouts, Simulation Demo	Quiz, Exam, Assignment
CO5	Design Op-Amp Circuits to perform arithmetic operations on Analog Signals, e.g, Summing, Subtracting, Exponentiation, etc.	PO3	Cognitive/Create,	Lectures, Notes/Handouts,	Quiz, Q/A, Exam, Assignment
CO6	Prototype electronic circuits on breadboards using basic electrical equipment (such as DC-Supply, Function-Generators, etc.) and, match their behavior with theoretical models	PO5	Cognitive/Create, Analyze, Psychomotor/Precision, Manipulation	Lab Class	Lab Work, Lab Test
CO7	Use circuit simulation tools to design complicated circuits and perform different types of analyses (DC, AC, Transient, etc.) to predict their behavior before implementing them in real life	PO5	Cognitive/Analyze, Apply, Evaluate Psychomotor/Precision,	Simulation Demo	Lab Test

F. Course Materials:

i. Text and Reference Books:

Sl.	Title	Author(s)	Publication Year	Edition	Publisher	ISBN
1	Microelectronic Circuits	Adel S. Sedra, Kenneth C. Smith	2015	7th ed.	Oxford University Press	978–0–19–933913–6
2	Foundations of Analog and Digital Electronic Circuits	Anant Agarwal, Jeffrey H. Lang	2005	1st ed.	Morgan Kaufmann Publishers	978–1–55–860735–4
3	The Art of Electronics	Paul Horowitz, Winfield Hill	2016	3rd ed.	Cambridge University Press	978–0–521–80926–9
4	Operational Amplifiers and Linear Integrated Circuits	Robert F. Coughlin, Frederick F. Driscoll	2001	6th ed.	Prentice Hall	978–0–130–14991–6

ii. Other materials

a. Lecture Notes/Handouts
b. Video Lectures (EdX Course – Circuits and Electronics)
c. Lab Sheets
d. Simulation Tool (EveryCircuit)
e. Simulation Tool (PSPICE)

G. Lesson Plan:

Theory

No	Topic	Week/Lecture#	Related CO (if any)
	History and Importance of Electronic Devices – Diodes, Electronic Switches and Amplifiers, Transition from Mechanical Switches to Vacuum Tubes to Solid State Devices, Current State-of-the-Art in Electronics/Semiconductor Technology	Lecture-1	CO1
	Introduction to I-V Characteristics. I-V Characteristics of: Simple Linear Elements – Resistors, Voltage Source and Current Source; Hybrid Linear Elements – Voltage	Lecture-2	CO1

	Source in series with Resistor, Current Source in parallel with Resistor; Source-Conversion		
	I-V Characteristics of: Degenerate Elements – Open-Circuit and Short-Circuit; Finding I-V of Series/Parallel Combination I-V Characteristics of Piecewise Linear Elements. Finding Device Parameters from I-V Graphs, Problem-Solving	Lecture-3	CO1
	Solving Circuits with Piece-Wise Linear Elements – Method of Assumed States, The Mili-Kilo convention, Load-Line	Lecture-4	CO1
	Introduction to Non-Linear Elements – *Ideal* Diode. Advantages/Uses of Non-Linear Elements - Half-Wave and Full-Wave Rectifiers	Lecture-5	CO2
	Constructing a Real Diode – Introduction to Semiconductors, n-type and p-type doping, P-N junction, Schokley Diode Equation	Lecture-6	CO1
	PWL Model of Diodes, Solving Diode Circuits, Diode Logic Gates.	Lecture-7	CO1
	Rectifiers Revisited – Average Value of Output, Smoothing Capacitor, Peak-to-Peak Ripple, Ripple Factor	Lecture-8	CO2
	Clipper and Clamper Circuits	Lecture-9	CO2
	Introduction to Voltage Regulators, Zener Diodes, Breakdown Voltage. Finding V_{in}(min) and I_{out}(max) of an *ideal* Zener Regulator.	Lecture-10	CO2
	Regulators Revisited, Line-Regulation, Load Regulation, Knee Current, Problem-Solving	Lecture-11	CO2
	Biasing, Diode Review	Lecture-12	CO1
Mid-Term Exam			
	Introduction to Electronic Switches, Basic Inverter, Diode-Transistor Logic Introduction to Amplifiers and Controlled Sources, Input/Output Resistance, Voltage-Current Conversion using Resistors, BJT and FET	Lecture-13	CO3
	Operation of an Ideal BJT- Cut-Off, Active and Saturation Mode. Transfer Characteristics, Inverting Amplifier	Lecture-14	CO3

	Operation of an Ideal FET- Cut-Off, Saturation and Triode Mode. Transfer Characteristics, Inverting Amplifier	Lecture-15	CO3
	Ideal Output Characteristics, Solving Transistor Circuits using Method of Assumed States, Problem-Solving	Lecture-16	CO3
	Constructing a *real* BJT – npn and pnp transistors. Ebers-Moll Equation, PWL Model & Non-ideal Analysis	Lecture-17	CO1,3
	Constructing a *real* MOSFET – n/p-channel, enhancement/depletion-type MOSFETs. Output Characteristics, PWL Model and Non-ideal Analysis	Lecture-18	CO1,3
	Transistor Circuits with Emitter/Source Resistance, Common Emitter/Source Amplifier (Follower/Buffer), Negative-Feedback, Common-Base/Common-Gate Amplifier	Lecture-19	CO4
	Solving Transistor Circuits with Emitter/Source Resistance, Loaded Amplifiers, Cascaded/Multi-Stage Amplifiers	Lecture-20	CO4
	Introduction to Operational Amplifiers – Differential Amplifiers, Op-Amp Circuits in Open-Loop Configuration – Square Wave Generator, Characteristics of Infinite Gain, Op-Amp Circuits in Closed Loop Configuration - Controlling Gain through Negative Feedback, Virtual Ground	Lecture-21	CO5
	Op-Amp Circuits in Closed Loop Configuration – Inverting Amplifier, Non-Inverting Amplifier, Follower, Buffer, Inverting Weighted Summer, Weighted Subtractor, Exponential Converter, Logarithmic Converter, Multiplier, Divider, Differentiator, Integrator	Lecture-22	CO5
	Op-Amp Problem-Solving	Lecture-23	CO5
Final Exam			

<u>LAB</u>

No	Topic	Week/Lecture#	Related CO (if any)
	Study Of Diode Characteristics	Lab-1	CO6
	Study Of Half-Wave And Full-Wave Diode Rectifier Circuit	Lab-2	CO6
	Study Of Clipper And Clamper Circuits	Lab-3	CO6

	Study Of Zener Diode And Its Application In Voltage Regulation	Lab-4	CO6
	Simulating Diode Circuits in PSPICE	Simulation-1	CO7
Hardware Lab Test			
	Study Of Bipolar Junction Transistor Characteristics	Lab-5	CO6
	Study Of Common Emitter (CE) Amplifier	Lab-6	CO6
	Study Of I-V Characteristics Of MOSFET	Lab-7	CO6
	Simulating Transistor Circuits in PSPICE	Simulation-2	CO7
Software Lab Test			

H. Assessment Tools:

Assessment Tools	Weightage (%)
Class Performance, Attendance and Question/Answer	10
Problem Sets/Assignment	5
Quizzes	20
Midterm Exam	20
Final Exam	25
Lab Work (Hardware & Software)	20

I. CO Assessment Plan:

Assessment Tools	Course Outcomes						
	CO1	CO2	CO3	CO4	CO5	CO6	CO7
Quizzes	√	√	√	√	√		
Assignments	√	√			√		
Midterm Exam	√	√	√				
Final Exam		√	√	√	√		
Lab Work						√	√

Digital Logic Design

A. Course General Information:

Course Code:	CSE260
Course Title:	Digital Logic Design
Credit Hours (Theory+Lab):	3 + 0
Contact Hours (Theory+Lab):	3 + 3
Category:	Program Core
Type:	Required, Engineering, Lecture + Laboratory
Prerequisites:	None
Co-requisites:	None

B. Course Catalog Description (Content):

This course provides an introduction to digital systems such as computers, communication and information systems. Firstly, the course will cover Boolean algebra, digital logic gates, combinational logic circuits, decoders, encoders, multiplexers and demultiplexers. The course will then cover sequential circuits: asynchronous and synchronous counters, registers, flip-flops. An introduction to memory elements and registers will also be provided. Hands-on experience will be provided through lab works and lab project. The course includes a compulsory 3-hour laboratory work each week.

C. Course Objective:

The objectives of this course are to

a. Familiarization with different number systems and conversion
b. Introduce Boolean logic operation and teach students how to use Boolean Algebra and K-maps to realize two-level minimal/optimal combinational circuits
c. Teach students other Boolean simplification methods such as Tabulation
d. Expose students in the introductory design process of combinational circuits including MSI circuits
e. Teach basic operation and analysis of sequential circuits using latches, flip-flops, counters, registers and memory elements.

D. Course Outcomes (COs):

Upon successful completion of this course, students will be able to

Sl.	CO Description	Weightage (%)
CO1	**Use** the knowledge of Boolean algebra to solve boolean equations and build circuits	10%
CO2	**Use** the knowledge of minimization techniques such as Karnaugh map and tabulation method to realize minimal/optimal combinational circuits along with calculating SOP and POS	25%
CO3	**Design** and **analyze** different combinational and sequential circuits such as adder, comparator, encoder, decoder, multiplexer, demultiplexer, counter, register, memory, etc.	40%
CO4	**Operate** laboratory equipment build, and troubleshoot simple combinational and sequential circuits.	15%
CO5	**Work** in a team and communicate effectively	5%

E. Mapping of CO-PO-Taxonomy Domain & Level- Delivery-Assessment Tool:

Sl.	CO Description	POs	Bloom's taxonomy domain/level	Delivery methods and activities	Assessment tools
CO1	**Use** the knowledge of Boolean algebra to solve boolean equations and build circuits	**c**	Cognitive	Lectures, notes, lab class	Quiz, Exam
CO2	**Use** the knowledge of minimization techniques such as Karnaugh map and tabulation method to realize minimal/optimal combinational circuits along with calculating SOP and POS	**c**	Cognitive	Lectures, notes, lab class	Quiz, Exam
CO3	**Design** and **analyze** different combinational and sequential circuits such as adder, comparator, encoder, decoder, multiplexer, demultiplexer, counter, register, memory.	**c**	Cognitive	Lectures, notes, lab class	Quiz, Exam, Design Project
CO4	**Operate** laboratory equipment build, and troubleshoot simple combinational and sequential circuits	**e**	Psychomotor	Lab Class	Lab Work, Design Project
CO5	**Work** in a team and communicate effectively	i	Affective	Lab Class	Lab work, Design Project

F. Course Materials:

i. Text and Reference Books:

Sl.	Title	Author(s)	Publication Year	Edition	Publisher	ISBN
1	Digital Design	M Morris Mano & M D Ciletti	2012	5^{th} ed.	Pearson Education	ISBN-13: 978-0-13-277420-8
2	Digital Design: Principles and Practices	J F Wakerly	2005	4^{th} ed.	Prentice Hall	ISBN-13: 978-0131863897

ii. Other materials (if any)

a. Lecture notes and presentation slides
b. Lab hand-outs
c. Lab usage manual

G. **Lesson Plan:**

No	Topic	Week/Lecture#	Related CO (if any)
	Introduction, review of number systems, and binary arithmetic	Lecture 1-2	
	Introduction to boolean algebra and simplification	Lecture 3	CO1
	SOP, POS and boolean functions minimization techniques (K-map)	Lecture 4-8	CO2
	Boolean functions minimization techniques (Tabulation)	Lecture 9-10	CO2
	Midterm		
	Combinational circuit design process using adder and subtractor	Lecture 12-13	CO3
	Combinational circuit design using Decoder, Encoder, Multiplexer	Lecture 14-15	CO3
	Introduction to sequential circuit design process (Flip Flop)	Lecture 16-17	CO3
	Sequential circuit analysis	Lecture 18-20	CO3
	Counter and Register	Lecture 21-22	CO3

	Review	Lecture 23-24	
Final Exam			

H. Assessment Tools:

Assessment Tools	Weightage (%)
Class Performance & Attendance	5
Assignment	10
Quizzes	15
Midterm Exam	20
Final Exam	30
Project work	5
Lab work (Hardware)	15

I. CO Assessment Plan:

Assessment Tools	Course Outcomes			
	CO1	CO2	CO3	CO4
Quizzes (Q)	x	x		
Homework	x	x		
Midterm exam	x			
Final Exam	x	x		
Project work			x	x
H/W Lab work			x	x

Data Communications

A. Course General Information:

Course Code:	CSE 320
Course Title:	Data Communication
Credit Hours (Theory):	3+0
Contact Hours (Theory):	3
Category:	Core
Type:	Lecture
Prerequisites:	None

B. Course Catalog Description (Content):

This course will present an introduction to purpose and methods of communication. Necessity for modulation and techniques. Technical aspects of data communications. Effects of noise and control. Basic concepts such as fundamental limits, encoding, modulation, multiplexing, error detection and control. Topics include: Data Transmission Protocols, different layers in data communication systems, LANs, WANs linked with telephony.

C. Course Objective

a. Introduce Students with fundamental concepts of data communication
b. Teach students about purpose and methods of communication
c. Teach necessity for modulation, encoding, multiplexing technique
d. Explain effects of noise, error detection and control

D. Course Outcomes (COs):

Upon successful completion of this course, students will be able to

Sl.	CO Description	Weightage (%)
CO 1	**Describe** the elements of data communication with different network topologies and the functionality of each protocol layer of two network models: OSI and TCP/IP	20
CO 2	**Understand** the basics of signal, bit and performance measurements and different signal conversion techniques based on advantages and disadvantages.	30
CO 3	**Decide** which bandwidth utilization technique to use in a practical scenario: Multiplexing approaches.	20
CO 4	**Analyze** the major components of telephone and cable networks and different transmission mediums based on their physical properties in data transmission.	10
CO 5	**Identify** link layer concepts, protocols, and services such as error detection and **show** multi-access techniques	20

E. Mapping of CO-PO-Taxonomy Domain & Level- Delivery-Assessment Tool:

Sl.	CO Description	POs	Bloom's taxonomy domain/level	Delivery methods and activities	Assessment tools
CO1	**Describe** the elements of data communication with different network topologies and the functionality of each protocol layer of two network models: OSI and TCP/IP	a	Cognitive/Analyze	Lectures, notes	Quiz, exam
CO2	**Understand** the basics of signal, bit and performance measurements and different signal conversion techniques based on advantage and disadvantages	c	Cognitive/Create	Lectures, notes	Quiz, exam
CO3	**Decide** which bandwidth utilization technique to use in a practical scenario: Multiplexing approaches.	d	Cognitive/Analyze	Lectures, notes	Quiz, exam
CO4	**Analyze** the major components of telephone and cable networks and different transmission mediums based on their physical properties in data transmission.	d	Cognitive/Analyze	Lectures, notes	Quiz, exam
CO5	**Identify** link layer concepts, protocols, and services such as error detection and multi-access techniques.	a	Cognitive/Analyze	Lectures, notes	Quiz, exam

F. Course Materials:

i. Text and Reference Books:

Sl.	Title	Author(s)	Publication Year	Edition	Publisher	ISBN
1	Data Communication and Networking	Behrouz A. Forouzen	2017	5^{th}	McGraw Hill	ISBN 10: 1259064751
2	Data and Computer Communications	William Stallings	2013	10^{th}	Pearson	ISBN-10: 0133506487

ii. Other materials (if any)

Lecture Notes and presentation slides

G. Lesson Plan:

No	Topic	Week/Lecture#	Related CO (if any)
1	Introduction to elements of data communication, Network topology, Protocols and standards, Network models: OSI and TCP/IP model, Physical and logical addressing.	**Week 1,2**	CO1
2	Basics concepts of signal: Analog and digital signal and their properties, Transmission impairment, Data rate limit calculation	**Week 3**	CO2
3	Digital to Digital conversion- Line coding, Block coding and Scrambling techniques	**Week 4**	CO2
4	Analog to Digital conversion techniques- PCM, DM.	**Week 5**	CO2
Mid Exam (Week 7)			
5	Bandwidth utilization: Multiplexing and spreading, FDM, WDM, Synchronous Time-Division Multiplexing, FHSS and Direct sequence spread spectrum.	**Week 6,8**	CO3
6	Major components of telephone network, Dial-Up modems and modem standards, Brief	**Week 8**	CO4

	idea of DSL, Cable TV network for data transfer- Downstream and Upstream data band, Downstream and Upstream sharing, Guided and Unguided Medium- Twisted-Pair, Coaxial and fiber optic cable, radio and microwaves, wireless media.		
7	Data link layer concepts, services and multi-access protocols: Channel Partitioning, Random Access and Taking Turns protocols.	**Week 9,10**	CO5
8	Different types of error detection and correction mechanisms in the Data Link layer: Block coding, Hamming distance, CRC, Checksum.	**Week 11,12**	CO5
Final Exam			

H. Assessment Tools:

Assessment Tools	**Weightage (%)**
1. Participation in class	5 %
2. Quizzes/Class Tests	15%
3. Mid Term Examination	25 %
4. Assignments	15 %
5. Final	40 %

I. CO Assessment Plan:

Assessment Tools	**CO1**	**CO2**	**CO3**	**CO4**	**CO5**
Homework		x			x
Quizzes	x	x	x		
Examinations	x	x	x	x	x

Operating Systems

A. Course General Information:

Course Code:	CSE321
Course Title:	Operating Systems
Credit Hours (Theory+Lab):	3 + 0
Contact Hours (Theory+Lab):	3 + 3
Category:	Program Core
Type:	Required, Engineering, Lecture
Prerequisites:	CSE221: Algorithms
Co-requisites:	None

B. Course Description:

Principles of operating systems: design objects; sequential process; concurrent processes, functional mutual exclusion, processor co-operation and deadlocks, management. Control and scheduling of large information processing systems. Dispatching processor access methods, job control languages memory addressing, paging and store multiplexing, and time sharing, batch processing. Scheduling algorithms, file systems, and security; semaphores and critical sections, device drivers, multiprocessing, sharing, design and implementation methodology, performance evaluation and case studies. The course includes a compulsory 3 hour laboratory work each week.

C. Course Outcomes:

CO 1	**Review** the fundamentals concepts of computer system organization and the structure of operating systems.
CO 2	**Examine** various aspects of process management in operating system
CO 3	**Knows** how different CPU scheduling algorithm works and their respective importance
CO 4	**Explains** the concept of threads and different kinds of threads

CO 5	**Inspect** process synchronization mechanisms and deadlocks
CO 6	**Problem Analysis** the management of main and virtual memory
CO 7	**Assess** disk scheduling algorithms and storage management methods
CO 8	**Perform** a case study of existing desktop operating systems

D. Mapping of CO-PO-Taxonomy Domain & Level- Delivery-Assessment Tool:

Sl.	CO Description	POs	Bloom's taxonomy domain/level	Delivery methods and activities	Assessment tools
CO 1	**Review** the fundamentals concepts of computer system organization and the structure of operating systems.	**PO1**	Cognitive/Understanding	Discussion, Q/A.	Quiz, Exam, Assignment
CO 2	**Examine** various aspects of process management in operating system	**PO2**	Cognitive/ Analyzing	Lectures, Discussion, Q/A, Lab tasks	Quiz, Exam, Assignment
CO 3	**Analyze** how different CPU scheduling algorithm works and their respective importance	**PO4**	Cognitive/ Analyzing	Lectures, Discussion, Q/A, Lab tasks	Quiz, Exam
CO 4	**Explains** the concept of threads and different kinds of threads	**PO1**	Cognitive/Understanding	Lectures, Discussion, Q/A, Lab tasks	Quiz, Assignment
CO 5	**Inspect** process synchronization mechanisms and deadlocks	**PO5**	Cognitive/ Analyzing	Lectures, Discussion, Q/A, Lab tasks	Quiz, Exam, Assignment
CO 6	**Problem Analysis** the management of main and virtual memory	**PO2**	Cognitive/ Analyzing	Lectures, Discussion, Q/A.	Quiz, Exam,
CO 7	**Assess** disk scheduling algorithms and storage management methods	**PO2**	Cognitive/ Understanding	Lectures, Discussion, Q/A	Quiz, Exam,
CO 8	**Perform** a case study of existing desktop operating systems	**PO4**	Cognitive/ Creating	Lectures, Discussion, Q/A	Quiz, Exam,

E. Course Outlines:

Sr. No.	Topic details	Time allocation
1	Concepts, history of operating system, computer system architecture. Services provided by the operating systems, operating system structure, storage structure, protection and security and system call.	**Week 1,2**
2	Process concepts, states, scheduling, Inter Process Communication (IPC), operation on process (creation and termination).	**Week 3**
3	Basic concepts of CPU scheduling, scheduling criteria, scheduling algorithms and simulations (FCFS, SJF, Priority, Round Robin), and Multilevel queue and Multilevel feedback queue scheduling.	**Week 4,5**
4	Threads overview, Multicore Programming, Multithreading Models, Thread Libraries, Threading Issues and operating systems examples.	**Week 6**
5	Process Synchronization, Basic Concepts, The Critical-Section Problem , Peterson's Solution, Synchronization Hardware, Semaphores and Classic problems of synchronization (Bounded Buffer, Reader-Writer, Dining-Philosopher etc.). Concepts of deadlock, resource allocation graph and wait for graph, methods of deadlock handling (deadlock prevention, avoidance and detection and recovery).	**Week 7,8,9**
6	Memory management Background, Swapping, Memory Allocation (MFT, MVT, Paging, Segmentation). Introduction of virtual memory, demand paging, performance of demand paging, page fault handling, page replacement algorithms (FIFO, LRU, Optimal) and thrashing.	**Week 10, 11**
7	Overview of Mass storage structure, disk scheduling (FCFS, SSTF, SCAN, C-SCAN, C-LOOK)	**Week 12, 13**
8	Perform a case study of existing desktop operating systems (Windows, Linux and Android etc.).	**Week 14**

F. Assessment Methods vs. Course Outcomes:

Assessment Methods	CO 1	CO 2	CO 3	CO 4	CO 5	CO 6	CO 7	CO 8
Homework	X	X		X	X			
Quizzes		X	X		X			
Examinations	X	X	X	X	X	X	X	X

Laboratory Works			X		X	X	X	
Case Study					X			X

G. Textbook:

1	Operating System Concepts Essentials by Abraham Silberschatz, Peter B. Galvin and Greg Agne, Wiley; 10th Edition (July 5, 2008). ISBN-10: 0470128720
2	Modern Operating Systems by Tanenbaum A.S., Prentice Hall; 3rd Edition (2007). ISBN-13: 978-0136006633

H. Course Assessment Methods:

Guidelines for CSE course teaching in BRAC University. The following assessment methods are based on a Theory Course with Lab.

Section	**Marks (%)**
Participation in class	5 %
Quizzes/Class Tests/Assignments/	10 % - 15 %
Mid Term Examination	20 % - 25 %
Lab & Project	20 % - 30 %
Final	30 % - 40 %
Total	**100 %**

Numerical Methods

A. Course General Information:

Course Code:	CSE330
Course Title:	Numerical Methods
Credit Hours (Theory+Lab):	3+0
Contact Hours (Theory+Lab):	3+3
Category:	Program Core
Type:	Required
Prerequisites:	MAT216
Co-requisites:	None

B. Course Catalog Description (Content):

Computer Arithmetic: floating point representation of numbers, arithmetic operations with normalized floating point numbers; Iterative methods: different iterative methods for finding the roots of an equation and their computer implementation; Solution of simultaneous Algebraic Equations, Gauss elimination; Interpolation, Least square approximation of functions, Taylor series representation, Chebyshev series; Numerical differentiation and integration and Numerical Solution of Differential Equations.

C. Course Objective:

The course will help students to

- recognize the need for numerical analysis, and the importance of error analysis.
- learn various methods to linearize a polynomial, differentiate and integrate different functions by using approximations,
- learn how to solve the linearized equation by using the laws of linear algebra, like Gaussian elimination, QR decomposition, etc.

D. Course Outcomes (COs):

Upon successful completion of this course, students will be able to

Sl.	CO Description	Weightage (%)
CO1	**Demonstrate** an understanding of the fundamental concept of numerical analysis and the need for error analysis.	10
CO2	**Interpolate** polynomial functions, test their convergence, examine interpolation error.	20
CO3	**Differential** by finite difference method and estimate rounding error, solve numerically the nonlinear equations by various methods, like Newton's method, etc. Numerical integration by Newton-Cotes method.	40
CO4	**Solve** linear equations by Gaussian method, QR decomposition.	30

E. Mapping of CO-PO-Taxonomy Domain & Level- Delivery-Assessment Tool:

Sl.	CO Description	POs	Bloom's taxonomy domain/level	Delivery methods and activities	Assessment tools
CO1	Demonstrate an understanding of the fundamental concept of numerical analysis and the need for error analysis.	PO2	Cognitive/Analyze	Lectures, Notes, Lab	Assignments, Quiz, Midterm, Final exam
CO2	Interpolate polynomial functions, test their convergence, examine interpolation error	PO3	Cognitive/Evaluate	Lectures, Notes, Lab	Assignments, Quiz, Midterm, Final exam
CO3	Differential by finite difference method and estimate rounding error, solve numerically the nonlinear equations by various methods, like Newton's method, etc. Numerical integration by Newton-Cotes method.	PO1	Cognitive/Analyze	Lectures, Notes, Lab	Assignments, Quiz, Midterm, Final exam
CO4	Solve linear equations by Gaussian method, QR decomposition.	PO1	Cognitive/Analyze	Lectures, Notes, Lab	Assignments, Quiz, Midterm, Final exam

F. Course Materials:

i. Text and Reference Books:

Sl.	Title	Author(s)	Publication Year	Edition	Publisher	ISBN
1.	Numerical Analysis II – Lecture Notes. (Main Text)	Anthony Yeates.	2018		Durham University	
2.	Numerical Methods In Engineering With Python.	Jaan Kiusalaas.	2005	First Edition	Cambridge University Press	ISBN-13: 978-0-521-85287-6 ISBN-10:0-521-85287-0
3.	Numerical Analysis (Reference)	Richard L. Burden J. Douglas Faires	2011	Ninth Edition	BROOKS/COLE CENGAGE Learning	ISBN-13:978-0-538-73351-9 ISBN-10:0-538-73351-9

ii. Other materials (if any)

(a) Lecture Notes/Handouts

G. **Lesson Plan:**

No	Topic	Lecture#
1	Introduction. Overview of the whole course	1
2	Fixed-point arithmetic	2
3	Rounding error, loss of significance	3
4	Polynomial interpolation, Lagrange form	4
5	Newton difference form, interpolation error	5
6	Convergence, derivative conditions	6
7	Differentiation: higher order finite difference	7
8	Rounding error, Richardson extrapolation	8
9	Nonlinear equations: Interval bisection, Fixed point iteration	9

10	Orders of convergence, Newton's method	10
11	Aitken acceleration, Quasi-Newton methods	11
Midterm Exam: Tentatively March 9th, 2022. It will be confirmed later.		
12	Linear equations: Triangular system, Gaussian elimination	12
13	LU decomposition, Pivoting	13
14	Vector norms, Matrix norms	14
15	Conditioning, Iterative methods	15
16	Least Square Approximation: Orthogonality	16
17	Discrete least squares, QR decomposition	17
18	Continuous least squares, orthogonal polynomials	18
19	Newton-Cotes formulae	19
20	Composite Newton-Cotes formulae	20
21	Exactness	21
22	Gaussian quadrature	22
Final Exam (Central) : April 30, 2022 (Saturday) between 8am-11am.		

H. Assessment Tools:

Assessment Tools	**Weightage (%)**
Attendance	10
Quiz	10
Assignment	10
Midterm	20
The mandatory final exam	30
Lab	20

I. CO Assessment Plan:

Assessment Tools	Course Outcomes			
	CO1	**CO2**	**CO3**	**CO4**
Assignments	√	√	√	√
Quiz	√	√	√	√
Mid Term	√	√	√	√
Final	√	√	√	√

Automata and Computability

A. Course General Information:

Course Code:	CSE331
Course Title:	Automata and Computability
Credit Hours:	3
Contact Hours:	3
Category:	Core
Prerequisites:	CSE221

B. Course Catalog Description (Content):

An introduction to finite representation of infinite objects and basic mathematical models of computation. Finite automata and regular languages, pushdown automata and context free languages. Turing machines. Church's Thesis. Partial recursive functions. Undecidability. Reducibility and completeness. Halting problem. Time complexity and NP-completeness. Probabilistic computation. Interactive proof systems.

C. Course Objective:

The objectives of this course are to

01. To make students exPOre the fundamental capabilities and limitations of computer algorithms, according to various computational models and measures.
02. To equip students with the prerequisite knowledge to write automated text processing programs like lexers and parsers of compilers.

D. Course Outcomes (COs):

Upon successful completion of this course, students will be able to

Sl.	CO Description
CO1	**Demonstrate** mastery over regular languages, regular expressions, and deterministic finite automata.
CO2	**Demonstrate** mastery over context-free languages, context-free grammars, and pushdown automata.
CO3	**Apply** the abstract mathematical reasoning skills learned elsewhere.
CO4	By learning about regular expressions and context-free grammars, gain the prerequisites for writing many text processing programs including but not limited to lexers and parsers of compilers.

I. Mapping of CO-PO-Taxonomy Domain & Level- Delivery-Assessment Tool:

Sl.	CO Description	POs	Bloom's taxonomy domain/level	Delivery methods and activities	Assessment tools
CO1	**Demonstrate** mastery over regular languages, regular expressions, and deterministic finite automata.	**PO1**	Cognitive/Understanding	Discussion, Q/A.	Quiz, Exam, Assignment
CO2	**Demonstrate** mastery over context-free languages, context-free grammars, and pushdown automata.	**PO2**	Cognitive/ Analyzing	Lectures, Discussion, Q/A, Lab tasks	Quiz, Exam, Assignment
CO3	**Apply** the abstract mathematical reasoning skills learned elsewhere.	**PO4**	Cognitive/ Analyzing	Lectures, Discussion, Q/A, Lab tasks	Quiz, Exam
CO4	**Experimenting** with many text processing programs including but not limited to lexers and parsers of compilers.	**PO3**	Cognitive/Applying	Lectures, Discussion, Q/A, Lab tasks	Quiz, Assignment

E. Course Materials:

i. Text and Reference Books:

Sl.	Title	Author(s)	Publication Year	Edition	Publisher	ISBN
01	Introduction to Automata Theory, Languages, and Computation.	John E. Hopcroft, Rajeev Motwani, Jeffrey D. Ullman.	2006	3rd	Prentice Hall	
02	Introduction to the Theory of Computation	Michael Sipser	2013	3rd	Cengage	

F. Assessment Tools:

Assessment Tools	Weightage (%)
Attendance	10%
Assignments (written+pdf submission)	15%
Quizzes (written+pdf submission, best 2/4 counted)	20%
Midterm (on campus)	25%
Final (on campus)	30%

G. Lesson Plan:

No	Topic	Week/Lecture#
1	Regular languages, regular expressions, deterministic finite automata.	Week 1-3
Quiz #1, 23-02-2022		

2	Non-deterministic finite automata, equivalence of NFA and DFA, DFA minimization.	Week 4-5
Quiz #2, 03-03-2022		
3	Equivalence of regular expressions and deterministic finite automata.	Week 6
Midterm (10-03-2022), 5:00 pm.		
4	Introduction to context-free languages, CFG, derivation, parse trees, and ambiguity.	Week 7-8
Quiz #3, 06-04-2022		
5	Chomsky Normal Form Cocke-Younger-Kasami algorithm.	Week 9
6	Pushdown automata.	Week 10
Quiz #4, 20-04-2022		
7	Pushdown automata (continued) and review.	Week 11-12
Final Exam, 08-05-2022, 5:00 pm.		

Computer Architecture

A. Course General Information:

Course Code:	CSE 340
Course Title:	Computer Architecture
Credit Hours (Theory + Lab):	3+0
Contact Hours (Theory + Lab):	3+0
Category:	Program Core
Type:	Required, Engineering, Lecture
Prerequisites:	CSE260
Co-requisites:	None

B. Course Catalog Description (Content):

A systematic study of the various elements in computer design, including circuit design, storage mechanisms, addressing schemes, and various approaches to parallelism and distributed logic. Information representation and transfer; instruction and data access methods; the control unit; hardware and micro programmed; memory organization; RISC and CISC machines.

C. Course Objective:

The objectives of this course are:

a) Introduce different processor technologies, performance metrics, representation of numbers and arithmetic operations.
b) Introduce MIPS architecture, demonstrate its instruction formats, their data path designing process and translation of simple C/Java code snippets to MIPS assembly language.
c) Teach how to design processor datapaths and recognize pipelining hazards and different techniques for overcoming them.
d) Introduce and explain memory hierarchy and performance analysis.
e) Introduce parallel architecture and parallel programming.

D. Course Outcomes (COs):

Upon successful completion of this course, students will be able to

Sl.	CO Description	Weightage (%)
CO1	**Explain** the processor technologies and performance metrics alongside the representation of numbers and arithmetic operations	25
CO2	**Demonstrate** various instruction formats, their encoding, translation from C/Java code to MIPS instruction	25
CO3	**Visualize** the datapath of different instructions and recognize various pipelining hazards and hazard overcoming techniques	20
CO4	**Outline** memory hierarchy and performance analysis and **review** various parallel architecture and its programming	10

E. Mapping of CO-PO-Taxonomy Domain & Level- Delivery-Assessment Tool:

Sl.	CO Description	POs	Bloom's taxonomy domain/level	Delivery methods and activities	Assessment tools
CO1	**Explain** the processor technologies, performance metrics alongside the representation of numbers and arithmetic operations	a	Cognitive	Lectures, slides, notes	Midterm, Final
CO2	**Demonstrate** various instruction formats, their encoding, translation from C/Java code to MIPS instruction	a	Cognitive	Lectures, slides, notes	Midterm, Final
CO3	**Visualize** the datapath of different instructions and **recognize** various pipelining hazards and hazard overcoming techniques	b	Cognitive	Lectures, slides, notes	Midterm, Final
CO4	**Outline** memory hierarchy and performance analysis and **review** various parallel architecture and its programming	a	Cognitive	Lectures, slides, notes	Midterm, Final

F. Course Materials:

i. Text and Reference Books:

Sl.	Title	Author(s)	Publication Year	Edition	Publisher	ISBN
1	Computer Organization and Design: The Hardware/Software Interface	D. A. Patterson, J. L. Hennessy	2013	5th Edition	Morgan Kaufmann	978-0124077263
2	Computer Architecture: A Quantitative Approach	D. A. Patterson, J. L. Hennessy	2017	6th Edition	Morgan Kaufmann	978-0128119051
3	Computer Architecture and Organization	J. P. Hayes	1997	3rd Edition	McGraw Hill	0-07-027366-9

ii. Other materials (if any)

Lecture Notes and other material will be made available on buX.

G. Lesson Plan:

No	Topic	Week/Lecture #	Related CO (if any)
1	Various Computer Architecture Models, Measuring Performance CPI, Performance Equation, RISC and CISC Architecture	1,2	CO1
2	Introduction of the Computer Architecture, Introduction to MIPS Assembly Language, Instruction sets, Conversion from High-level code to MIPS equivalent form, how function call works, etc.	2, 3, 4, 5, 6	CO2
3	Arithmetic for Computers: Addition, Subtraction, Multiplication, Division, IEEE-754 floating-point conversion, floating-point operations in MIPS architecture	6, 7	CO1
Midterm			
4	MIPS non-pipelined datapath/control path, how single-cycle datapath works for various instruction, advantages, disadvantages of single-cycle datapath	8	CO3
4	MIPS pipelined datapath, how it works, advantages and disadvantages, designing pipelined datapath, Hazard and hazard overcoming techniques	9	CO3

5	Memory Hierarchy, Memory Technology, Cache Introduction and Improving Cache Performance, Virtual memory, Cache Miss	10, 11	CO4
6	Introduction of Parallel Processors, SISD, MIMD, SIMD, SPMD, and Vector, Multicore, multi-threading, GPU	12, 13	CO4
Final			

H. Assessment Tools:

Assessment Tools	**Weightage (%)**
1. Participation in class	5
2. Quizzes/Class Tests	15
3. Assignments	15
4. Mid Term Examination	25
5. Final Examination	40
Total	100%

I. CO Assessment Plan:

Assessment Tools	**Course Outcomes**			
	CO1	**CO2**	**CO3**	**CO4**
1. Quizzes/Class Tests				
2. Mid-Term Examination	√	√		
3. Assignments				
4. Final Exam	√	√	√	√

Microprocessors

A. Course General Information:

Course Code:	CSE 341
Course Title:	Microprocessors
Credit Hours (Theory + Lab):	3+0
Contact Hours (Theory + Lab):	3+3
Category:	Program Core
Type:	Required, Engineering, Lecture + Laboratory
Prerequisites:	CSE 260: Digital Logic Design
Co-requisites:	None

B. Course Catalog Description:

Introduction to different types of microprocessors, their architectures and their functionalities. This course focuses on 8086 microprocessor's architecture, instruction set, interfacing I/O devices, interrupt structure, DMA, ICs for interfacing. Later, some advanced microprocessor concepts of microprocessor based system design are introduced. The course includes a compulsory 3 hour laboratory work each week.

C. Course Objective

The objectives of this course are to:

a. Getting familiar with the architecture of Intel 8086 microprocessor and the associated components of a computer system.

b. Introducing the instruction set of 8086 microprocessor and different addressing modes of this microprocessor.

c. Explain the techniques used in microprocessors to interface various devices to it.

d. Teach the students about interrupts in 8086 and how interrupts are handled.

e. Describe the advantage of direct memory access approach and how DMA controller 8237 circuit works.

f. Introducing later processors (80x86) and comparing the features and components of them.

g. Understand assembly language, develop assembly language programs and learn how to interface them to various devices using the hardware kit.

D. Course Outcomes (COs):

Upon successful completion of this course, students will be able to

Sl.	CO Description	Weightage (%)
CO1	**Illustrate** the Architecture of Intel 8086 microprocessor and the components of a computer system.	15
CO2	**Explain** the instruction set and addressing modes of 8086 microprocessor.	10
CO3	**Illustrate** the ways to interface various devices to the microprocessor.	10
CO4	**Explain** interrupts and interrupt handling in 8086 microprocessor.	10
CO5	**Describe** direct memory access approach and DMA controller 8237 circuit.	10
CO6	**Compare** the features and components of 80X86 processors.	20
CO7	**Develop** programs using assembly language programs and learn how to interface them to various devices using the hardware kit.	25

E. Mapping of CO-PO-Taxonomy Domain & Level- Delivery-Assessment Tool:

Sl.	CO Description	POs	**Bloom's taxonomy** domain/level	Delivery methods and activities	Assessment tools
CO1	Illustrate the Architecture of Intel 8086 microprocessor and the components of a Computer system.	PO1	Cognitive/ Understand	Lectures, Discussion, Q/A, Lab Tasks	Quiz, Exams, Assignments
CO2	Explain the instruction set and addressing modes of 8086 microprocessor.	PO2	Cognitive/ Understand	Lectures, Discussion, Q/A, Lab Tasks	Quiz, Exams, Assignments
CO3	Illustrate the ways to interface various devices to the microprocessor.	PO1	Cognitive/ Understand	Lectures, Discussion, Q/A, Lab Tasks	Quiz, Exams

CO4	Explain interrupts and interrupt handling in 8086 microprocessor.	PO2	Cognitive/ Understand	Lectures, Discussion, Q/A	Quiz, Exams, Assignments
CO5	Describe direct memory access approach and DMA controller 8237 circuit.	PO1	Cognitive/ Understand	Lectures, Discussion, Q/A	Quiz, Exams
CO6	Compare the features and components of 80X86 processors.	PO1	Cognitive/ Analyze	Lectures, Discussion, Q/A	Quiz, Exams
CO7	Develop programs using assembly language programs and learn how to interface them to various devices using the hardware kit.	PO5	Cognitive/ Understand, Psychomotor/ Precision	Lectures, Discussion, Q/A, Lab Tasks	Laboratory works, Project & viva

F. Course Materials:

i. Text and Reference Books:

Sl.	Title	Author(s)	Publication Year	Edition	Publisher	ISBN
1.	Microprocessors and Interfacing	Douglas V. Hall	1992	2nd Edition	McGraw Hill	ISBN: 978-0071004626
2.	Assembly Language Programming Organization of the IBM PC	Ytha Yu, Charles Marut	1992	International Ed.	McGraw Hill	ISBN: 978-0070726925
3.	Intel Microprocessors, Architecture, Programming and Interfacing	Barry B. Brey	2009	8th Edition	Pearson Prentice Hall	ISBN–13: 978–0–13–502645–8

ii. Other materials (if any)

a. Lecture notes and presentation slides
b. Lab hand-outs
c. Lab usage manual
d. Emulator (emu8086)

H. Lesson Plan for Laboratory:

No	Topic	Week/Lecture #	Related CO (if any)
1.	Introduction to the basics of a microprocessor and intel8086 chipset, along with the emu8086, the simulator used to code in assembly language, representation of numbers and characters in assembly language.	Week 1	CO1, CO7
2.	Explaining the basic idea on the registers involved in 8086 processors, how the registers can be used as variables and special features of some registers, performing basic arithmetic operations.	Week 2	CO1, CO7
3.	Introduction to Interrupts for basic input output mechanisms.	Week 3	CO7
4.	Familiarize the students with how decisions can be made with the jump instruction, conditional and unconditional jump, implement high-level language decision structures using assembly.	Week 4	CO7
5.	Introduction to loop instructions, different looping structures.	Week 5	CO7
6.	Make students acquainted with how one-dimensional arrays are declared in assembly language along with the operations they are used in and how array elements can be accessed by using different addressing modes.	Week 6	C02, CO7
7.	Learning basic push pop operations using stack and solving different problems in a more effective way using stack.	Week 7	CO7
8.	Introduce the students to get the basic idea of Macros and Procedures and understand how they work.	Week 8	CO7
9.	Introduction to the basics of an EMU/MDA8086 Trainer Kit and its components. Hands on knowledge on how to configure the kit using "Machine Code" mode.	Week 9	CO7
10.	Final Assessment (viva and project)	Week 10, 11	-

I. Assessment Tools:

Assessment Tools	Weightage (%)
1. Participation in class	5 %
2. Quizzes & Assignments	15 %
3. Mid Term Examination	25 %
4. Lab Works & Final Assessment	25 %

5. Final	30 %
Total	**100 %**

J. CO Assessment Plan:

Assessment Tools	**Course Outcomes**						
	CO1	**CO 2**	**CO3**	**CO4**	**CO 5**	**CO 6**	**CO 7**
Assignment	√	√		√			
Quizzes	√	√	√	√	√	√	
Midterm Examination	√	√	√				
Final Examination	√	√	√	√	√	√	
Laboratory Works	√	√					√
Final Assessment (viva & project)							√

Digital Electronics and Pulse Techniques

A. Course General Information:

Course Code:	CSE 350
Course Title:	Digital Electronics and Pulse Techniques Digital Electronics and Pulse Techniques Laboratory
Credit Hours (Theory+Lab):	3 + 0
Contact Hours (Theory+Lab):	3 + 3
Category:	Program Core
Type:	Required, Engineering, Lecture + Laboratory
Prerequisites:	CSE260 Digital Logic Design, CSE251 Electronic Devices and Circuits
Co-requisites:	None

B. Course Catalog Description (Content):

Diode logic gates, transistor switches, transistor gates, MOS gates, Logic families: TTL, ECL, IIL and CMOS logic with operation details. Propagation delay, product and noise immunity. Open collector and High impedance gates. Electronic circuits for flip flops, counters and register, memory systems. PLA's (A/D, D/A converters with applications, S/H circuits) LED, LCD and optically coupled oscillators. Non-linear applications of OPAMPs. Analog switches. Linear wave shaping: diode wave shaping techniques, clipping and clamping circuits, comparator circuits, switching circuits. Pulse transformers, pulse transmission. Pulse generation:monostable, bistable and stable multivibrations, Timing circuits. Simple voltage sweeps, linear circuit sweeps. Schmitrigger, blocking oscillators and time base circuit. The course includes a compulsory 3 hour laboratory work each week.

C. Course Objective:

The objectives of this course are to

a. Familiarization with different logic families of circuits
b. Teach students how to use basic electronics to realize combinational digital circuits
c. Teach students how to build simple memory and computational unit for digital system
d. Expose students in the pulse wave shaping and oscillators using different electronic circuit elements
e. Introduce students to design pulse generation in different ways and means of transmission

D. Course Outcomes (COs):

Upon successful completion of this course, students will be able to

Sl.	CO Description	Weightage (%)
CO1	**Understand** the switching operation of transistors and related concepts	15%
CO2	**Design** and **Analyze** electronic circuit level implementation of a combinational logic function for different logic families	25%
CO3	**Design** the registers and memory elements using electronic components, PLAs (A/D, D/A converters with applications, S/H circuits)	10%
CO4	**Design** pulse shaping techniques using op-amp and diodes	10%
CO5	**Design** pulse transformers, pulse transmission system, pulse generation system using mono-stable, bi-stable and stable multi-vibrations and different time based circuit	20%
CO6	**Operate** laboratory equipment build, and troubleshoot simple combinational and sequential electronic circuits	20%

E. Mapping of CO-PO-Taxonomy Domain & Level- Delivery-Assessment Tool:

Sl.	CO Description	POs	Bloom's taxonomy domain/level	Delivery methods and activities	Assessment tools
CO1	**Understand** the switching operation of transistors and related concepts	**PO3**	Cognitive/ Apply	Lectures, notes, lab class	Quiz, Exam
CO2	**Design** and **Analyze** electronic circuit level implementation of a combinational logic function for different logic families	**PO4**	Cognitive/ Apply	Lectures, notes, lab class	Quiz, Exam, Design Project
CO3	**Design** the registers and memory elements using electronic components, PLAs (A/D, D/A converters with applications, S/H circuits)	**PO3**	Cognitive/ Create	Lectures, notes, lab class	Quiz, Exam
CO4	**Design** pulse shaping techniques using op-amp and diodes	**PO3**	Cognitive/ Create	Lectures, notes, lab class	Quiz, Exam

CO5	**Design** pulse transformers, pulse transmission system, pulse generation system using mono-stable, bi-stable and stable multi-vibrations and different time based circuit	**PO3**	Cognitive/ Create	Lectures, notes, lab class	Quiz, Exam, Design Project
CO6	**Operate** laboratory equipment build, and troubleshoot simple combinational and sequential electronic circuits	**PO5**	Cognitive/ Remember, Psychomotor / Precision	Lectures, lab class	Lab work, Design Project

F. Course Materials:

i. Text and Reference Books:

Sl.	Title	Author(s)	Publication Year	Edition	Publisher	ISBN
1	Electronic Circuits Analysis and Design	Donald A Neamen	2008	3rd ed.	McGraw-Hill	ISBN-13: 978-0-07-352958-5
2	Integrated Electronics: Analog and Digital Circuits and System	Jacob Millman, Christos C. Halkias	2010	2nd ed.	McGraw-Hill	ISBN-13: 978-0070151420

ii. Other materials:

a. Lecture notes and presentation slides
b. Lecture videos and tutorials
c. Lab hand-outs
d. Lab usage manual

G. Lesson Plan:

No	Topic	Lectures	Related CO
1	Introduction, what is digital electronics and pulse techniques, development of digital electronic, different type of logic gates, fan-in, fan-out, types of delay, noise margin, power dissipation, speed power product	Lecture 1-2	CO1

2	Review of diode, transistor and CMOS, identify different types of logic gates (diode logic, resistor transistor logic and transistor transistor logic), introduction to resistor transistor logic (RTL) gates	Lecture 3-4	CO1
3	How diode transistor logic (DTL) gate work in different logic condition, calculation of noise margin and fan out, wired AND logic, high threshold logic	Lecture 5-6	CO2
4	Transistor transistor logic (TTL) NAND gate operation, totempole output, power dissipation, noise margin, fan-out calculation	Lecture 7-8	CO2
5	How emitter coupled OR/NOR gate work, designing ECL gate, fan out and power dissipation calculation.	Lecture 9-10	CO2
6	Different types of CMOS gates, CMOS logic design and stick diagrams, R-S flip, flop, CMOS transmission gate, full adder, shift register, memory, programmable logic array	Lecture 11-12	CO3
Midterm			
7	R-2R ladder, binary weighted digital to analog converter, resolution, full scale output, flash ADC, successive approximation and dual slope ADC,	Lecture 15-16	CO3
8	Series clipper, shunt clipper, Zener clipper, positive and negative clamping circuit	Lecture 17-18	CO4
9	Bi stable, mono stable and astable multivibrator, output waveshapes of different multivibrator, Operation of Pulse transformers, pulse transmission	Lecture 19-20	CO5
10	Schmitt trigger, Wave Generators: square wave, triangular wave and sawtooth wave generator.	Lecture 20-21	CO5
11	Review	Lecture 23-24	
Final Exam			

H. Assessment Tools:

Assessment Tools	Weightage (%)
Class Performance & Attendance	5
Assignment	5
Quizzes	15
Midterm Exam	20
Final Exam	35

Lab work	20

I. CO Assessment Plan:

Assessment Tools	Course Outcomes					
	CO1	**CO2**	**CO3**	**CO4**	**CO5**	**CO6**
Quizzes (Q)	✓	✓		✓	✓	
Assignment			✓	✓		✓
Midterm exam	✓	✓	✓			
Final Exam	✓	✓		✓	✓	✓
Lab work				✓	✓	

Computer Interfacing

A. Course General Information:

Course Code:	CSE360
Course Title:	Computer Interfacing
Credit Hours (Theory+Lab):	3 + 0
Contact Hours (Theory+Lab):	3 + 1.5
Category:	Hardware
Type:	Required
Prerequisites:	CSE341 Microprocessors

B. Course Catalog Description (Content):

This course will give an overview of computer interface components and their characteristics. It will cover detailed discussion on some programmable interfacing ICs, sensor, hardware, and software calibration on sensor interfacing, some processing devices which can interface computers with the real world and some output devices. This course also includes high power interface devices, LED, LCD, Seven segment display, transducers, stepper motors and peripheral devices. In addition, it will also introduce some software simulation tools to develop an interfacing system.

C. Course Objective:

a) Introduction with basic Interfacing, Interfacing Components and their characteristics.
b) Outline the working mechanism of different types of sensors, sensing devices, hall-effect sensing, and real life application of some sensors.
c) Describe various I/O interfacing including GPIO using MCU/MPU development board and how Programmable Peripheral Interface (PPI) – 82C55 works.
d) Explain the working mechanism of Disk, Drum, Motors and Printers and their real life application.
e) Describe the types and formats of communication protocols like I2C, SPI, UART, USART, and USB.
f) Explain LCD, LED, Seven- segment display, Keyboard, Mouse work and interface with their interfacing principal

D. Course Outcomes (COs):

Upon successful completion of this course, students will be able to

Sl.	CO Description	Weightage (%)
CO1	**Recall** interfacing, interfacing components and their characteristics, high-end and low-end interfacing and bus interfacing with necessary block diagrams.	15
CO2	**Understand** different types of Communication protocols and memory management for better communication between processor and sensor ICs.	10
CO3	**Demonstrate** working mechanism of various peripheral devices with Modern Development Boards and IC 82C55.	15
CO4	**Illustrate** digital system using various GPIOs such as sensors, display devices such as LCD, LED, Seven- segment display, Keyboard, Mouse etc. **Application** of the GPIOs in real-life systems.	20
CO5	**Adapt** some advanced interfacing concept using sensors to develop a computer interfacing project for real life problem solving in a group project.	10

E. Mapping of CO-PO-Taxonomy Domain & Level- Delivery-Assessment Tool:

Sl.	CO Description	POs	Bloom's taxonomy domain/level	Delivery methods and activities	Assessment tools
CO1	**Explain** the interface of various systems, sensing devices, low level/ high level interfacing, bus interfacing and types of interfacing in real life application.	a	Cognitive/ Understand	Lecture	Quiz, Mid
CO2	**Describe** computer interfacing and communication standards using I2C, SPI, UART, USART, and different memory types for better communication between processor and sensor ICs, handled with interrupt.	b	Cognitive/ Understand	Lecture	Quiz, Final
CO3	**Illustrate the working** mechanism of various peripheral devices with Modern Development Boards. Demonstrate the work of IC 82C55/	c	Cognitive / Analyze	Lecture	Quiz, Mid, Final
CO4	**Design** digital system using various interface modules, sensors, display devices such as LCD, LED, Seven- segment display, Keyboard, Mouse etc. **Application** of the GPIOs in real-life systems.	c	Cognitive/ Apply	Lecture	Quiz, Mid, Final
CO5	**Adapt** some advanced interfacing concepts using sensors to develop a computer interfacing project for real life problem solving in a group project.	i	Psychomotor, Affective	Lab	Lab Project

F. Course Materials:

i. Text and Reference Books:

Sl.	Title	Author(s)	Publication Year	Edition	Publisher	ISBN
1	Microprocessor Architecture Programming and Applications with the 8085	Ramesh S. Gaonkar	1984	5th	Penram International Publishing Private Limited	81-87972-09-2
2	"Introduction to ARM(R) Cortex Microcontroller"	Jonathan Valvano,	2014	5th	Createspace (2012)	978-147750899 2
3	Embedded Microcomputer Systems: Real Time Interfacing	Jonathan Valvano	2014	3rd	Cengage learning	13-978-1-111-42 625- 5

ii. Other materials (if any)

Lectures available at bux.

In addition, there are:

1) Presentation slides
2) Word Files
3) Book Chapter pdf
4) Videos and links

G. Lesson Plan (Theory):

No	Topic	Week/Lecture#	Related CO (if any)
1	Introduction to Computer interfacing: a) Introduction to Digital Systems b) Introduction to Computer Architecture c) Block Diagram of input output Interfacing d) Introduction to the system bus e) Introduction to low level program execution	Week 1	CO1
2	Interfacing and communication standards: a)BUS Interface: SCSI, USB, SCSI, 1394, ATA, SATA, PCI	Week 2, 3	CO1
3	Computer interfacing ICs: a) Basic I/O (Memory mapper and Isolated) and diagrams. b) Introduction to 82C55 programmable interfacing IC	Week 4, 5	CO3

	c) Pin configuration and Block diagram d) Control word of 82C55 e) Time diagram of handshaking input and output		
4	Display devices: a) LED and interfacing b) LCD and interfacing c) 7 segment display and interfacing multiple 7 segment display and interfacing	Week 6	CO4
5	Output devices: a) Different types of Motor Characteristics b) Different types of Actuator Characteristics c) Device Driver Circuits d) Switch, Matrix Keyboard (GPIO)	Week 7, 8	CO4
6	Sensing techniques and sensors a) Necessity and applications of sensor b) Physical properties of sensing (optics, sound, temperature, encoder etc.) c) Types of sensors (active, passive, simple, complex etc.) d) Sensor fusion e) Sensor circuit and hardware calibration f) Integration and software calibration of sensor g) Advance sensors	Week 8, 9	CO4
7	Processing devices: a) Introduction to Microprocessor, Micro- Controller, Arduino, Raspberry-PI, FPGA. b) Processor architectures	Week 10	CO4
8	Interfacing Communication Standards: a) Synchronous and Asynchronous transmission b) SPI, I2c, UART for better communication between processor and sensor ICs	Week 11, 12	CO2
9	Memory Management Systems: a) Differentiating between memories (Magnetic disk, SSD) b) Properties and working mechanisms of Magnetic disk and SSD	Week 13	CO2

Lesson Plan (Lab):

No	Topic	Week/Lecture# (Alternate Week)	Related CO (if any)
1	Introduction to Interfacing lab and equipments	Week 1	
2	Hands on Hardware Lab Task 1: Interfacing Output device (LED, Buzzer) and Input device (switch) with arduino uno and coding	Week 2	
3	Hands on Hardware Lab Task 2: Interfacing DHT11 sensor and LCD with arduino uno and coding	Week 3	
4	Final Hardware Project proposal submission with at least 2 sensors	Week 4	CO5
5	Project Update and Solving any project related problems	Week 5	CO5
6	Final project Showcase	Week 6	CO5

H. Assessment Tools:

Assessment Tools	Weightage (%)
1. Attendance	5 %
2. Quizzes	15 %
3. Mid Term Examination	20 %
4. Assignment	10 %
5. Final	30%
6. Lab	20%
Total	**100 %**

I. CO Assessment Plan:

Assessment Tools	Course Outcomes				
	CO1	**CO2**	**CO3**	**CO4**	**CO5**
Quizzes	X	X	X	X	
Mid	X	X	X	X	
Final		X	X	X	
Project and Term Paper					X

Database Systems

A. Course General Information:

Course Code:	CSE370
Course Title:	Database Systems
Credit Hours (Theory+Lab):	3+0
Contact Hours (Theory+Lab):	3+3
Category:	Program Core
Type:	Required, Engineering, Lecture + Laboratory
Prerequisites:	CSE221 Algorithms
Co-requisites:	None

B. Course Catalog Description (Content):

This course is designed as an introduction to relational database management systems (RDBMS) focusing on the efficient design, implementation and optimization of an RDBMS. Topics covered will include the advantages and disadvantages of DBMS, database architecture, data modeling using ER and EER models, relational integrity constraints, relational schema mapping from ER/EER, indexing, hashing and normalization. SQL Query formulation will be extensively practiced in both the theoretical and laboratory components of the course. The course includes a compulsory 3 hour laboratory work each week as CSE370L. Students must complete several hands-on SQL assignments and a group project for the laboratory work. The group project will involve the design and implementation of a complete database system including a user interface.

C. Course Objective:

The objectives of this course are to:

a. explain the advantages and disadvantages of using a DBMS over a file-based system
b. explain the process of data modeling using ER and EER models
c. discuss the relational integrity constraints and how they are enforced in a database system
d. teach the process of mapping an ER/EER model to a relational schema
e. introduce indexing and hashing and discuss their role in efficient data retrieval
f. explain the concepts of functional dependencies (FD) and normalization, and how they are used to optimize database design
g. teach students the methods for storing, manipulating and retrieving data using structured query language (SQL)

h. expose students to the process of implementing a complete database system project in a team using software tools, programming languages and SQL

D. Course Outcomes (COs):

Upon successful completion of this course, students will be able to

Sl.	CO Description	Weightage (%)
CO1	**Explain** fundamental concepts related to database management and **Identify** different database concepts discussed in a given scenario.	5
CO2	**Construct** appropriate ER/EER models to represent complex data requirements of an organization or system.	20
CO3	**Apply** schema design principles for mapping ER/EER models to relational schemas.	20
CO4	**Apply** normalization techniques to reduce data redundancy and optimize database schema design.	15
CO5	**Prepare** and **Implement** standard queries using Structured Query Languages (SQL) to store, retrieve and manipulate data	15
CO6	**Apply** appropriate indexing and hashing techniques to optimize database performance.	10
CO7	**Develop** a database application using SQL and other effective programming languages as a group project to solve a complex data management problem.	15

E. Mapping of CO-PO-Taxonomy Domain & Level- Delivery-Assessment Tool:

Sl.	CO Description	POs	Bloom's taxonomy domain/level	Delivery methods and activities	Assessment tools
CO1	**Explain** fundamental concepts related to database management and **Identify** different database concepts discussed in a given scenario.	**PO1 (a)**	Cognitive/Understand	Lectures, Notes, Classwork	Assignment/Quiz, Exam
CO2	**Construct** appropriate ER/EER models to represent complex data requirements of an organization or system.	**PO1 (a)**	Cognitive/Apply	Lectures, Notes, Practice Sheets, group activity for project	Exam, Assignment/Quiz, Project

CO3	**Apply** schema design principles for mapping ER/EER models to relational schemas.	**PO1 (a)**	Cognitive/Apply	Lectures, Notes, Practice Sheets, group activity for project	Exam, Assignment/Quiz, Project
CO4	**Apply** normalization techniques to reduce data redundancy and optimize database schema design.	**PO1 (a)**	Cognitive/Apply	Lectures, Notes, Practice Sheets	Exam, Assignment/Quiz
CO5	**Prepare** and **Implement** standard queries using Structured Query Languages (SQL) to store, retrieve and manipulate data	**PO5 (e)**	Cognitive/Apply Psychomotor/ Manipulation	Lab Work, Lectures, Practice Sheets	Assignment/Quiz, Exam, Project
CO6	**Apply** appropriate indexing and hashing techniques to optimize database performance.	**PO1 (a)**	Cognitive/Apply	Lectures, Notes, Practice Sheets	Assignment/Quiz, Exam
CO7	**Develop** a database application using SQL and other appropriate programming languages as a group project to solve a complex data management problem.	**PO3 (c)**	Cognitive/Create Psychomotor/ Manipulation	Lab Class	Project

F. Course Materials:

i. Text and Reference Books:

Sl.	Title	Author(s)	Publication Year	Edition	Publisher	ISBN
1	Fundamentals of Database Systems	Ramez Elmasri, Shamkant B. Navathe	2015	7th ed.	Pearson	**ISBN-13:** 978-0133970777
2	Database systems : a practical approach to design, implementation, and management	Thomas M. Connolly, Carolyn E. Begg	2014	6th ed.	Pearson	**ISBN-13:** 978-0132943260
3	Database Systems Concept	Silberschatz, Korth, Sudarshan	2011	6th ed.	McGraw-Hill	**ISBN:** 978-0-07-352332-3

ii. Other materials:

a. Lecture slides
b. Lab handouts
c. Command Line tool (MySQL Mini Server)

G. Lesson Plan (Theory):

No	Topic	Week/Lecture#	Related CO (if any)
1	Introduction - Fundamental Database System Concepts	Week 1/ Lecture 1-2	CO1
2	Data Modeling using the Entity-Relationship (ER) Model	Week 2/ Lecture 3-4	CO2
3	The Enhanced Entity-Relationship (EER) Model	Week 3/ Lecture 5-6	CO2
4	The Relational Data Model and Relational Database Constraints	Week 4/ Lecture 7-8	CO1
5	Review	Week 5	
MIDTERM - Week 6			
8	ER/EER to Relational Database Schema Mapping	Week 7-8/ Lecture 11 - 13	CO3
6	Indexing and Hashing	Week 8-9/ Lecture 14-16	CO6
7	Functional Dependencies and Normalization	Week 10/ Lecture 17-18	CO4
9	SQL Queries	Week 11/Lecture 19-20	CO5
10	Review	Week 12	
FINAL			

Lesson Plan (Laboratory):

No	Topic	Week/Lecture#	Related CO (if any)
	Introduction to MySQL: Environment setup, database and table creation, data insertion, alter table	Week 1	CO5
	Update and Delete queries, basic select queries: retrieve and sorting	Week 2	CO5
	Aggregate Functions, Nested and Sub queries, Grouping	Week 3	CO5
	Lab Assessment/Review	Week 4	
	Primary and Foreign keys, basic join queries	Week 5	CO5
	Advanced Join Queries	Week 6	CO5
	Lab Assessment/Review	Week 7	
	ER/EER Diagram and schema for group project	Week 8-9	CO2,CO3
	User Interface for group project and connection with MySQL using suitable programming language such as php/java/python	Week 10-11	CO5, CO7

	Project Demonstration	Week 12	CO7

H. Assessment Tools:

Assessment Tools	Weightage (%)
Class Performance and Attendance	5
Assignment	10
Quiz	10
Midterm	15
Final	30
Project	20
Lab work	10

I. CO Assessment Plan:

Assessment Tools	Course Outcomes						
	CO1	**CO2**	**CO3**	**CO4**	**CO5**	**CO6**	**CO7**
Assignment/Quiz	X	X	X	X		X	
Midterm	X	X					
Final			X	X	X	X	
Project		X	X		X		X
Lab work					X		

Compiler Design

A. Course General Information:

Course Code:	CSE420
Course Title:	Compiler Design
Credit Hours(Theory+Lab):	3+0
Contact Hours(Theory+Lab):	3+1.5
Category:	Program Core/Optional(CS)
Type:	Required,Engineering,Lecture+laboratory
Prerequisites:	CSE 221: Algorithms CSE 331: Automata and Computability
Co-requisites:	None

B. Course Catalog Description (Content):

Theory and Practice; An introduction to compiler and interpreter design, with emphasis on practical solutions using compiler writing tools such as Yacc in UNIX, and the C programming language, Topics covered include: lexical scanners, context free languages and pushdown automata, recursive descent parsing, bottom up parsing, attributed grammars, symbol table design, run time memory allocation, machine language, code generation and optimization.

C. Course Objective:

a. Introduce the fundamentals concepts of computer language processing system, different phases of a compiler, and analysis-synthesis model of compilation
b. Teach students regular expressions for recognizing tokens, building symbol table, and transforming the regular expressions to finite state machines for developing lexical analyzer.

c. Explain context-free-grammars and parsers for syntax analysis to validate string of tokens of a programming language.
d. Teach students to construct syntax-directed definitions, translation schemes, annotated parse trees, dependency graph for semantic analysis and type checking.
e. Help students to understand intermediate codes, and code optimization techniques for efficient compiler design.
f. Help students to learn about how to generate codes for target programming language considering memory, instruction cost, available registersand run-time memory allocation**.**

D. Course Outcomes (COs):

Upon successful completion of this course, students will be able to

Sl.	CO Description	Weightage (%)
CO 1	**Review** the fundamentals concepts of computer language processing system, different phases of a compiler, and analysis-synthesis model of compilation	5
CO 2	**Design** regular expressions for recognizing tokens, building symbol table, and transforming the regular expressions to finite state machines for developing lexical analyzer.	20
CO 3	**Design** context-free-grammars and parsers for syntax analysis to validate string of tokens of a programming language.	30
CO 4	**Construct** syntax-directed definitions, translation schemes, annotated parse trees, dependency graph for semantic analysis and type checking.	25
CO 5	**Illustrate** intermediate codes, and code optimization techniques for efficient compiler design and how to generate code for programming languages considering different cost	20

E. Mapping of CO-PO-Taxonomy Domain & Level- Delivery-Assessment Tool:

Sl.	CO Description	POs	Bloom's taxonomy domain/level	Delivery methods and activities	Assessment tools
CO1	**Review** the fundamentals concepts of computer language processing system, different phases of a compiler, and analysis-synthesis model of compilation	**PO1**	Cognitive/Understand	Lecture, notes	Quiz, Exam
CO2	**Design** regular expressions for recognizing tokens, building symbol table, and transforming the regular expressions to finite state machines for developing lexical analyzer.	**PO3**	Cognitive/Apply	Lecture, notes, lab	Quiz, Exam, lab
CO3	**Design** context-free-grammars and parsers for syntax analysis to validate string of tokens of a programming language.	**PO5**	Cognitive/Create	Lecture, notes, lab	Quiz, Exam, lab

CO4	**Construct** syntax-directed definitions, translation schemes, annotated parse trees, dependency graph for semantic analysis and type checking.	**PO4**	Cognitive/Apply	Lecture, notes, lab	Quiz, Exam, lab
CO5	**Illustrate** intermediate codes, and code optimization techniques for efficient compiler design and how to generate code for programming languages considering different cost	**PO3**	Cognitive/Understand	Lecture, notes, lab	Quiz, Exam, lab

F. Course Materials:

i. Text and Reference Books:

Sl.	Title	Author(s)	Publication Year	Edition	Publisher	ISBN
	Principles of Compiler Design	Alfred V. Aho, Jeffrey D. Ullman	1977	2nd	Addison-Wesley	0-32 1-4868 1 – 1
	Compiler Construction: Principles and Practice	Kenneth C. Louden	1997	1st	Cengage Learning	13: 978-0534939724

ii. Other materials (if any)

G. Lesson Plan:

No	Topic	Week/Lecture#	Related CO (if any)
	# Lecture 1: Introduction to Compilers **Review** the fundamentals concepts of computer language processing system, different phases of a compiler, and analysis-synthesis model of compilation	1st Week	CO1
	# Lecture 2: Lexical Analysis: Token, Pattern, Lexeme, Regular Expressions, DFA, NFA # Lecture 3: Lexical Analysis: RE to NFA, NFA to DFA using subset construction, NFA to CFG	2nd Week	CO2
	# Lecture 4: Lexical Analysis: RE directly to DFA by computing followpos		CO2
	# Lecture 5: Lexical Analysis: NFA to RE using Arden's Theorem, DFA Minimization using Hopcroft's Algorithm	3rd Week	CO2

	# Lecture 6: Practice on L-2, L-3, L-4 and L-5		CO2
	# Lecture 7: Syntax Analysis: Introduction to Parsing, Context Free Grammar, Parse tree	4th Week	CO3
	# Lecture 8: Syntax Analysis: Error Recovery, Ambiguous Grammar, Left Recursion Elimination, Left factoring		CO3
	# Lecture 9: Syntax Analysis: LL Grammar, Top down parsing, computing first and follow sets, Dealing with errors in Top down parsing, Panic and phrase level error recovery	5th Week	CO3
	# Lecture 10: Practice on L-7, L-8 & L-9		CO3
	# Lecture 11: Mid Term Examination	6th Week	CO3
	# Lecture 12: LR Grammar, Bottom up parsing algorithm		CO3
	# Lecture 13: LR(0) items, computing closure and goto, LR(0) automation, basic SLR parser	7th Week	CO3
	# Lecture 14: LR(1) items, LR(1) automation, LR(1) parsing table		CO3
	# Lecture 15: LALR parsing, using ambiguous grammar	8th Week	CO3
	# Lecture 16: Practice on L-12, L-13 & L-14		CO3
	# Lecture 17: Syntax Directed Translation: Semantic rules, synthesized and inherited attributes	9th Week	CO4
	# Lecture 18: Syntax Directed Translation: SDD vs SDT; SDD based on bottom up parsing, Annotated parse tree, Dependency graph, Evaluation order		CO4
	# Lecture 19: Syntax Directed Translation: SDD based on top down parsing	10th Week	CO4
	# Lecture 20: Practice on L-17, L-18, L-19		CO4
	# Lecture 21: Intermediate Code Generation: Linear and Graphical representation,	11th Week	CO5
	#Lecture 22: Three address code, DAG, Implementation of DAG, Value number method, Triple, Quadruple, Indirect triples		CO5

	# Lecture 23:Intermediate Code Generation: Translating Expressions, SDT into three address code, short circuit code	12th Week	CO5
	# Lecture 24: Intermediate Code Generation: SDT for flow of control statements, Generating for three address code for Booleans		CO5
	# Lecture 26: Type Checking, Run time environment, Code generation	13th Week	CO6

H. Assessment Tools:

Assessment Tools	Weightage (%)
1. Participation in class	5 %
2. Quizzes/Class Tests/Assignments/	10 % - 15 %
3. Mid Term Examination	20 % - 25 %
4. Lab & Project	20 % - 30 %
5. Final	30 % - 40 %
6. Participation in class	5 %

I. COAssessment Plan:

Assessment Tools	Course Outcomes					
	CO1	CO2	CO3	CO4	CO5	CO6
Homework/Assignment			X	X		
Quizzes	X	X	X		X	
Examinations	X	X	X	X	X	X
Laboratory Works		X	X			
Project						

Computer Networks

A. Course General Information:

Course Code:	CSE421/EEE465
Course Title:	Computer Networks
Credit Hours (Theory+Lab):	3+0
Contact Hours (Theory+Lab):	3+3
Category:	Program Core
Type:	Core
Prerequisites:	CSE 320,CSE221
Co-requisites:	None

B. Course Catalog Description (Content):

An introduction to the basics of transport connections and sessions. The protocol hierarchy, design issues in transport and session layer protocol, end-to-end protocols, message handling protocols, terminal and file transfer protocols, Internet TCP/IP protocols. End to end data networks, congestion control networks, wireless networks, mobile computing, high speed networks. Concurrent programming, data link layer, framing and error control, media access control. Models of distributed computation, management and resource control of networks and distributed operating systems, distributed file systems, caching scheduling, process migration. Fault tolerance, network security and privacy, algorithm for deadlock detection. Synchronization and concurrency control in distributed systems. The course includes a compulsory 3-hour laboratory work each week if the student wishes to obtain CCNA certification. Otherwise the course includes a compulsory 3-hour laboratory work alternate week.

C. Course Objective:

The objective of this course are to

a. Introduce the concepts of network architectures, topologies, layering and protocols.
b. Describe key application layer concepts such as network services required by applications, clients and servers.

c. Explain transport layer concepts, relationship with the network and application layers, and services such as principles of reliable data transfer and congestion.
d. Teach network layer concepts, routing principles, algorithms, and addressing and Internet's various protocols.
e. Identify link layer services, link layer address and multi-access techniques.
f. Teach basic knowledge of the use of cryptography and network security.
g. Explain the operation of wireless LANs based on the IEEE802.11 standards, and mobility.
h. Using simulation tools to observe and analyze behaviors of networking protocols.
i. Design and create a small network for an organization.

D. Course Outcomes (COs):

Upon successful completion of this course, students will be able to

Sl.	CO Description	Weightage (%)
CO1	**Describe** application layer concepts, protocol principles and network applications such as HTTP, SMTP, P2P and DNS protocols.	10%
CO2	**Break down** the principles and operation of transport layer protocols TCP and UDP, along with how TCP implements reliable data transfer and congestion control.	15%
CO3	**Interpret** network layer concepts, routing principles, algorithms, and addressing and operation of Internet protocols such as IPv4, DHCP, NAT, ICMP and IPv6.	30%
CO4	**Identify** link layer concepts, protocols and services such as error detection, addressing and multi-access techniques.	15%
CO5	**Explain** the operation of wireless LANs based on the IEEE802.11 standards, and mobility.	10%
CO6	Using networking simulation tools to **observe** and **analyze** behaviors of networking protocols.	5%
CO7	**Build** a small network for an organization.	15%

E. Mapping of CO-PO-Taxonomy Domain & Level- Delivery-Assessment Tool:

Sl.	CO Description	POs	Bloom's taxonomy domain/level	Delivery methods and activities	Assessment tools
CO1	**Describe** application layer concepts, protocol principles and network applications such as HTTP, SMTP, P2P and DNS protocols.	**PO1**	Cognitive/Understand	Lectures, Notes	Quiz, Exam
CO2	**Analyze** various principles and operation of transport layer protocols TCP and UDP, along with how TCP implements reliable data transfer and congestion control.	**PO2**	Cognitive/Analyze	Lectures, Notes	Quiz, Exam
CO3	**Interpret** network layer concepts, routing principles, algorithms, and addressing and operation of Internet protocols such as IPv4, DHCP, NAT, ICMP and IPv6.	**PO3**	Cognitive/Apply	Lectures, Notes	Quiz, Exam, Project
CO4	**Identify** link layer concepts, protocols and services such as error detection, addressing and multi-access techniques.	**PO1**	Cognitive/Understand	Lectures, Notes	Quiz, Exam
CO5	**Explain** the operation of wireless LANs based on the IEEE802.11 standards, and mobility.	**PO1**	Cognitive/Understand	Lectures, Notes	Quiz, Exam, Project
CO6	**Use** networking simulation tools to observe and analyze behaviors of networking protocols.	**PO5**	Cognitive/Psychomotor/Imitation	Lab Class	Lab Work, Project
CO7	Build a small network for an organization.	**PO5**	Cognitive/Psychomotor/Manipulation	Lab Class	Lab Work, Project
CO8	**Identify** the ethical issues, responsibilities and norms related to computer network system design	**PO8**	Cognitive/ Understand	Class discussion	Case-study report

F. Course Materials:

i. Text and Reference Books:

Sl.	Title	Author(s)	Publication Year	Edition	Publisher	ISBN
1	Computer Networking: A Top-Down Approach Featuring the Internet	Jim Kurose and Keith Ross	2016	7th	Pearson	ISBN-13: 978-0133594140 ISBN-10: 9780133594140
2.	Computer Networks – A Systems Approach	Larry L. Peterson and Bruce Davies	2011	5th	Morgan Kaufmann	ISBN-13: 978-0123850591 ISBN-10: 9780123850591

ii. Other materials (if any)

a. Lectures
b. Handouts
c. Cisco Online Curriculum Access

G. Lesson Plan:

No	Topic	Week/Lecture#	Related CO (if any)
	Introduce the concepts of network architectures, topologies, layering and protocols	Lecture 1	
	Key application layer concepts and protocols	Lecture 2-4	CO1
	Transport layer concepts and protocols	Lecture 5-8	CO2
	Network Layer Concepts	Lecture 9-12	CO3
Midterm			
	Network Layer protocols	Lecture 14-16	CO3
	Data Link layer concepts	Lecture 17-19	CO4
	Network Security	Lecture 20-21	CO5
	Wireless LANs and Mobility	Lecture 22-23	CO6

	Review	Lecture 24	
Final Exam			

H. Assessment Tools:

Assessment Tools	Weightage (%)
Class Performance and Attendance	5
Assignment	5
Quiz	10
Midterm Exam	20
Project	5
Lab Work	15
Final	40

I. CO Assessment Plan:

Assessment Tools			Course Outcomes					
	CO1	CO2	CO3	CO4	CO5	CO6	CO7	CO8
Quizzes	√	√	√	√	√	√		
Assignments	√		√	√	√	√		
Midterm Exam	√	√	√					
H/W Lab Work							√	√
S/W Lab Exam		√	√			√	√	
Project			√			√		√
Final Exam			√	√	√	√		

Artificial Intelligence

A. Course General Information:

Course Code:	CSE422
Course Title:	Artificial Intelligence
Credit Hours (Theory+Lab):	3 + 0
Contact Hours (Theory+Lab):	3 + 3
Category:	Program Core
Type:	Required, Theory, Lecture + Lab
Prerequisites:	CSE221: Algorithms
Co-requisites:	None

B. Course Catalog Description (Content): Survey of concepts in artificial intelligence. Knowledge representation, search and Control techniques. AI machines and features of LISP and PROLOG languages. Problem Representation; search, constraint propagation, rule chaining, frame inheritance, inference and learning in intelligent systems; systems for general problems solving, game playing, expert consultation, concept formation and natural languages processing; recognition, understanding and translation. Use of heuristic vs. algorithmic programming; cognitive simulations - vs. machine intelligence; study of some expert systems such as robotics and understanding. Solving problems in Al languages. The course includes a compulsory 3 hour laboratory work each week.

C. Course Objective:

1. Introducing the concept of Artificial Intelligence, rationality.
2. Enhance the understanding of different task environments and the appropriate use of different intelligent agents in these environments.
3. Analyze different problem solving strategies for informed, uninformed problems, deterministic or stochastic games, and constraint satisfaction problem.
4. Present different algorithms and the analysis of complexity, optimality and completeness of these algorithms.
5. Develop the critical skill to formulate problems and strategies to solve problems.
6. Using the concept of knowledge representation and logical inference in knowledge based agents.

7. Introduce the concept of uncertain knowledge and probabilistic reasoning.
8. Introduce probabilistic models so the students will be able to use these models in various decision making problems.
9. Introduce the basic concept of machine learning.

D. Course Outcomes (COs):

Upon successful completion of this course, students will be able to

	CO Description	Weightage (%)
	Identify properties of task environments and properties of intelligent agents. Describe problem structures and recognize algorithms that solve specific types of problems. Outline probabilistic models and their use. Know about the basics of machine learning; supervised learning and learning from decision trees, regression and classification with linear models.	15
	Explain different search strategies. Distinguish between search problems and constraint satisfaction problems. Distinguish between deterministic and stochastic environments and interpret the appropriate agents for knowledge representation and decision making. Distinguish between informed, uninformed and local search strategies. Generalize the structure of any problem. Illustrate different algorithms in different structures of problems. Select appropriate algorithms to solve problems. Identify supervised and unsupervised learning.	10
	Construct algorithms and apply these algorithms to solve generalized search problems and logical games. Construct problems and modify an algorithm to solve specific problems. Compute utility and predict results by using probabilistic theory. Moreover, apply these probabilistic theories like naïve Bayes theory and Bayesian networks in developing machine learning algorithms.	20
	Analyze and compare between the complexity, optimality and completeness of algorithms and other problem solving techniques. Compare between problem environments and select the appropriate solving strategy for the specific problem. Compare between knowledge representation by logical inference and probabilistic inference. Also, compare between different probabilistic theories and their complexities.	10
	Explain the concepts of Artificial Intelligence, rationality, task environment and agents. Assess different local search algorithms, adversarial search algorithms and constraint satisfaction problems. Interpret probabilistic theories, Bayesian inference and relate them to applications in machine learning.	15
	Combine the adversarial search algorithms to develop games. Generate solutions to constraint satisfaction problems. Solve informed search problems. Propose and plan projects to develop artificial intelligence based programs. Summarize current research on artificial intelligence and write reviews on them.	30

E. **Mapping of CO-PO-Taxonomy Domain & Level- Delivery-Assessment Tool:**

Sl.	CO Description	POs	Bloom's taxonomy domain/level	Delivery methods and activities	Assessment tools
1	**Identify** properties of task environments and properties of intelligent agents**. Describe** problem structures and **recognize** algorithms that solve specific types of problems. **Outline** probabilistic models and their use. **Know** about the basics of machine learning; supervised learning and learning from decision trees, regression and classification with linear models.	PO1	2	Lecture/Discussion/ Problem Solving Exercises/Lab Task	Quiz/Assignment/Midterm/Final/Attendance/Lab
		PO2	4		
2	**Explain** different search strategies. **Distinguish** between search problems and constraint satisfaction problems. **Distinguish** between deterministic and stochastic environments and interpret the appropriate agents for knowledge representation and decision making. Distinguish between informed, uninformed and local search strategies. **Generalize** the structure of any problem. **Illustrate algorithms** in different structures of problems. **Select** appropriate algorithms to solve problems. **Identify** supervised and unsupervised learning.	PO2	4	Lecture/Discussion/ Problem Solving Exercise/Lab Task	Quiz/Assignment/Midterm/Final/Lab
3	**Construct** algorithms and **apply** these algorithms to solve generalized search problems and logical games. **Construct** problems and **modify** an algorithm to solve specific problems. **Compute** utility and **predict** results by using probabilistic theory. Moreover, **apply** these probabilistic theories like Bayesian Inference and Naive Bayes theory in developing machine learning algorithms.	PO3	5	Lecture/Discussion/ Problem Solving Exercise/Lab Task	Assignment/ Lab work/Midterm/Final/Lab
		PO4	5		
		PO5	4		
4	**Analyze** and **compare between** the complexity, optimality and completeness of algorithms and other problem solving techniques. **Compare** between problem environments and **select** the appropriate solving strategy for any real life problems. **Compare** between knowledge representation by logical inference and probabilistic inference. Also, **compare** between different probabilistic theories and their complexities.	PO6	4	Lecture/Discussion/ Problem Solving Exercise	Quiz/Assignment/Midterm/Final
		PO2	5		

		PO3	5		
5	**Explain** the concepts of Artificial Intelligence, rationality, task environment and agents. **Assess** different local search algorithms, adversarial search algorithms and constraint satisfaction problems. I**nterpret** probabilistic theories and Bayesian networks and **relate** them to applications in machine learning.	PO2	5	Lecture/Discussion/ Problem Solving Exercise/Lab Task	Quiz/Assignment/Lab
		PO4	4		
6	**Combine** the adversarial search algorithms to develop games. **Generate** solutions to constraint satisfaction problems. **Solve** informed search problems. **Propose** and **plan** projects to **develop** artificial intelligence based programs. **Summarize** current research on artificial intelligence and write literature reviews on them.	**PO9**	4	Lab Work/ Research Work	Lab
		PO12	5		

F. Course Materials:

i. Text and Reference Books:

Sl.	Title	Author(s)	Publication Year	Edition	Publisher	ISBN
1	Artificial Intelligence A Modern Approach	Stuart Russel, Peter Norvig	1995	Third	Pearson Education	978-0-13-604259-4

G. Lesson Plan:

No	Topic	Week/Lecture#	Related CO (if any)
1	Introduction to Artificial Intelligence	Week 1/ Lecture 1	CO1
2	Intelligent Agents and Environment	Week 2/ Lecture 2, 3	CO1
3	Solving Problems by Searching	Week 3/ Lecture 4, 5	CO2, CO3, CO6

4	Beyond Classical Search 1	Week 4/ Lecture 6, 7	CO4, CO5, CO6
5	Beyond Classical Search 2	Week 5/ Lecture 8, 9	CO4, CO5, CO6
	Midterm		
6	Adversarial Search	Week 7/ Lecture 10, 11	CO4, CO5, CO6
7	Constraint Satisfaction Problem	Week 8/ Lecture 12, 13	CO4, CO5
8	Quantifying Uncertainty	Week 9/ Lecture 14, 15	CO1, CO4, CO5
9	Probabilistic Reasoning	Week 10/ Lecture 16, 17	CO3, CO5, CO6
10	Learning from Examples	Week 11/ Lecture 18, 19	CO1, CO6

(Lab component)

No	Topic	Week/Lecture#	Related CO (if any)
1	Solving Problems by Searching	Week 2/ Lecture 1	CO2, CO3, CO6
2	Beyond Classical Search	Week 3/ Lecture 2	CO5, CO6
3	Adversarial Search	Week 4/ Lecture 3	CO5, CO6
4	Machine Learning Basics	Week 5/ Lecture 4	CO1, CO2
	Midterm		
6	Data Cleaning/Processing	Week 7/ Lecture 5	CO1, CO2
7	Regression Analysis	Week 8/ Lecture 6	CO3, CO6
8	Learning from Examples	Week 9/ Lecture 7	CO3, CO5, CO6
9	Project/Presentation	Week 10, 11/ Lecture 8, 9	CO3, CO5, CO6

H. Assessment Tools:

Assessment Tools	**Weightage (%)**
Quiz	10
Assignment	10

Lab	25
Midterm	20
Final	30
Attendance	5

I. CO Assessment Plan:

Assessment Tools	Course Outcomes					
	CO1	CO2	CO3	CO4	CO5	CO6
Quiz	✔	✔		✔	✔	
Assignment	✔	✔	✔	✔	✔	
Lab	✔	✔	✔		✔	✔
Midterm	✔	✔	✔	✔		
Final	✔	✔	✔	✔		
Attendance	✔					

Computer Graphics

A. Course General Information:

Course Code:	CSE423
Course Title:	Computer Graphics Computer Graphics Laboratory
Credit Hours (Theory+Lab):	3 + 0
Contact Hours (Theory+Lab):	3 + 1.5
Category:	Core
Type:	Engineering, Lecture + Laboratory
Prerequisites:	-
Co-requisites:	MAT216

B. Course Catalog Description (Content):

Introduction to Graphical data processing. Fundamentals of interactive graphics Architecture of display devices and connectivity to a computer. Implementation of graphics concepts of two dimensional and three dimensional viewing, clipping and transformations. Hidden line algorithms. Raster graphics concepts: Architecture, algorithms and other image synthesis methods. Design of interactive graphic conversations. The course includes a compulsory 3 hour laboratory work alternate week.

C. Course Objective:

The objectives of this course are to

01. make students familiar with the basic steps of the graphics pipeline.
02. make students understand the theories and underlying mathematics of graphics applications.
03. understand the algorithms commonly used in 3D computer graphics
04. make the students understand advanced computer graphics techniques and applications
05. demonstrate how to solve problem by designing project

D. Course Outcomes (COs):

Upon successful completion of this course, students will be able to

	CO Description	Weightage (%)
	Demonstrate advanced knowledge on the fundamentals of 2D and 3D computer graphics	20%
	Explain and apply the algorithms commonly used in 3D computer graphics	30%
	Display competency in a number of advanced computer graphics techniques and applications	30%
	Design lab project	20%

E. Mapping of CO-PO-Taxonomy Domain & Level- Delivery-Assessment Tool:

Sl.	CO Description	POs	Bloom's taxonomy domain/level	Delivery methods and activities	Assessment tools
CO1	**Demonstrate** advanced knowledge on the fundamentals of 2D and 3D computer graphics	**PO1**	Cognitive/Apply	Lectures, notes	Quiz, Exam
CO2	**Explain** and apply the algorithms commonly used in 3D computer graphics	**PO1, PO5**	Cognitive/Apply Psychomotor/ Manipulation	Lectures, notes, Lab tasks	Assignment, Exam
CO3	**Display** competency in a number of advanced computer graphics techniques and applications	**PO1**	Cognitive/Apply	Lectures, notes, Problem solving exercises	Assignment, Quiz, Exam
CO4	**Design** Lab project	**PO9**	Cognitive/Creating	lecture, Discussion, Lab works	Project

F. Course Materials:

i. Text and Reference Books:

Sl.	Title	Author(s)	Publication Year	Edition	Publisher	ISBN
01	Computer Graphics: Principles and Practice	John F. Hughes, James D. Foley, Andries van Dam, Steven K. Feiner	1982	2nd	Pearson	ISBN-13: 978-0201848403 ISBN-10: 9780201848403

G. Lesson Plan:

No	Topic	Week/Lecture#	Related CO (if any)
	Shape drawing algorithms: Line, Circle, Ellipse, 8-way symmetry;	Week 1-2	CO1, CO2
	Clipping algorithm: Cohen-Sutherland	Week 3	CO1
	Clipping algorithm: Cyrus-Beck	Week 4	CO2, CO3
Midterm			
	Overview of Transformation, Modeling	Week 5	CO1
	Transformation in details	Week 6-7	CO1, CO2, CO3
	Projection	Week 8	CO1, CO2, CO3
	Color models, Illumination	Week 9	CO1, CO3
	Shading	Week 10	CO3
Final Exam			

H. Assessment Tools:

Assessment Tools	Weightage (%)
Attendance	5%
Assignments	5%

Quizzes	20%
Laboratory Work	20%
Midterm Examination	20%
Final Examination	30%

I. CO Assessment Plan:

Assessment Tools	Course Outcomes		
	CO1	**CO2**	**CO3**
Quizzes	√		√
Assignment/Homework			√
Midterm Exam	√	√	√
Final Exam	√	√	√
Project work and Lab work		√	

VLSI Design

A. Course General Information:

Course Code:	CSE460
Course Title:	VLSI Design VLSI Design Laboratory
Credit Hours (Theory+Lab):	3 + 0
Contact Hours (Theory+Lab):	3 + 3
Category:	Program Core
Type:	Required, Engineering, Lecture + Laboratory
Prerequisites:	CSE350 Digital Electronics and Pulse Techniques
Co-requisites:	None

B. Course Catalog Description (Content):

Design and analysis techniques fro VLSI circuits. Design reliable VLSI circuits, noise considerations, design and operation of large fan out and fan in circuits, clocking methodologies, techniques for data path and data control design. Simulation techniques. Parallel processing, spatial purpose architectures in VLSI. VLSI layouts partitioning and placement routing and wiring in VLSI. Reliability aspects of VLSI design. The course includes a compulsory 3 hour laboratory work alternate week. This course is accompanied by a mandatory 3 hour lab session each week.

C. Course Objective:

The objectives of this course are to

a. give a brief introduction to the complete process of chip designing
b. provide background knowledge to design, simulate and implement combinational & sequential CMOS circuits
c. teach theoretical concepts to critically analyze the cost & performance of an IC

d. expose the students to the fabrication process & physical design aspects of CMOS IC design
e. introduce the necessary programming language/softwares such as: Verilog (programming language), Quartus II, Microwind & DSCH2 to design, simulate, visualize and verify CMOS logic circuits and layouts

D. Course Outcomes (COs):

Upon successful completion of this course, students will be able to

Sl.	CO Description	Weightage (%)
CO1	**Understand** the complete lifecycle of an IC from silicon to chip	25%
CO2	**Design** combinational & sequential circuits in CMOS technology	10%
CO3	**Compute** current, voltage, power, delay & noise margin of CMOS logic circuits	20%
CO4	**Compare** cost, speed, area requirement & energy efficiency of different CMOS implementations	10%
CO5	**Optimize** circuit layout for better performance	10%
CO6	**Describe** how data is actually stored in hardwares at the device level	05%
CO7	**Simulate** combinational & sequential circuits elements using a Hardware Description Language (Verilog)	10%
CO8	**Design, Optimize & Verify** physical layout of an IC using DSCH2 & Microwind	10%

E. Mapping of CO-PO-Taxonomy Domain & Level- Delivery-Assessment Tool:

Sl.	CO Description	POs	Bloom's taxonomy domain/level	Delivery methods and activities	Assessment tools
CO1	**Understand** the complete lifecycle of an IC from silicon to chip	**PO1**	Cognitive/Understand	Lectures, Notes	Quiz, Exam
CO2	**Design** combinational & sequential circuits in CMOS technology	**PO3**	Cognitive /Create	Lectures, Notes	Quiz, Exam
CO3	**Compute** current, voltage, power, delay & noise margin of CMOS logic circuits	**PO2**	Cognitive/Apply	Lectures, Notes, Assignments	Quiz, Exam

CO4	**Compare** cost, speed, area requirement & energy efficiency of different CMOS implementations	**PO4**	Cognitive/Evaluate	Lectures, Notes, Assignments	Quiz, Exam
CO5	**Optimize** circuit layout for better performance	**PO4**	Cognitive/Apply	Lectures, Notes	Quiz, Exam
CO6	**Describe** how data is actually stored in hardwares at the device level	**PO1**	Cognitive/Remember	Lectures, Notes	Quiz, Exam
CO7	**Simulate** combinational & sequential circuits elements using a Hardware Description Language (Verilog)	**PO5**	Cognitive/Analyze, Psychomotor/Manipulation, Psychomotor/Precision	Lectures, Notes, Lab Experiments, Lab Assignments	Lab Exam, Viva
CO8	**Design, Optimize & Verify** physical layout of an IC using DSCH2 & Microwind	**PO3**	Cognitive/Create, Cognitive/Apply, Cognitive/Analyze, Psychomotor/Naturalization	Lectures, Notes, Lab Experiments, Lab Assignments	Lab Exam, Viva

F. Course Materials:

i. Text and Reference Books:

Sl.	Title	Author(s)	Publication Year	Edition	Publisher	ISBN
1	CMOS VLSI Design: A Circuits and Systems Perspective	Neil H. E. Weste & David Money Harris	2011	4th ed.	Addison-Wesley	ISBN 13: 978-0-321-54774-3
2	Fundamentals of Digital Logic with Verilog Design	Stephen Brown & Zvonko Vranesic	2014	3rd ed.	McGraw-Hill	ISBN 978–0–07–338054–4

ii. Other materials (if any)

a. Lecture notes and presentation slides
b. Class notes

c. Lab handouts and presentation slides
d. Softwares: Quartus II, DSCH2, Microwind (shall be provided at on-campus computer labs)

G. Lesson Plan:

i. Theory (2 lectures/week, 1 hour 20 minutes each)

No	Topic	Week/Lecture#	Related CO (if any)
1	Introduction to VLSI, Review of digital electronics	Week 1/Lecture 1-2	CO1
2	Introduction to CMOS circuits	Week 2/Lecture 3-4	CO2
3	Finite State Machines	Week 3-4/Lecture 5-8	CO2
4	Fabrication, Layout & Stick diagrams	Week 5/Lecture 9-10	CO1, CO4, CO5
5	CMOS Transistor Theory	Week 6/Lecture 11-12	CO3, CO4
Midterm (Week 7)			
6	DC Response, Power	Week 8/Lecture 13-14	CO3, CO4
7	Delay in CMOS circuit	Week 9-10/Lecture 15-18	CO3, CO4, CO5
8	Physical design	Week 11/Lecture 19-20	CO4, CO5
9	Memory elements	Week 12/Lecture 21-22	CO1, CO4
10	Review	Week 13/Lecture 23-24	
Final Exam (Week 14)			

ii. Lab (1 lecture/week, 2 hours 20 minutes each)

No	Topic	Week/Lecture#	Related CO (if any)
1	Experiment 0: Introduction to Verilog	Week 1-2/Lecture 1-2	CO7
2	Experiment 1: Verilog Design Part 1 (combinational logic)	Week 3/Lecture 3	CO7

3	Experiment 2: Verilog Design Part 2 (sequential logic)	Week 4/Lecture 4	CO7
4	Experiment 3: Verilog Design Part 3 (FSMs)	Week 5-6/Lecture 5-6	CO7
Midterm (Week 7)			
5	Experiment 4: DSCH2 Schematic Design Part 1	Week 8/Lecture 7	CO8
6	Experiment 5: DSCH2 Schematic Design Part 2	Week 8/Lecture 7	CO8
7	Experiment 6: Microwind Layout Design Part 1	Week 9/Lecture 8	CO8
8	Experiment 7: Microwind Layout Design Part 2	Week 9/Lecture 9	CO8
9	Review	Week 10, 11	
Final Exam (Week 12)			

H. Assessment Tools:

Assessment Tools	**Weightage (%)**
Class Performance & Attendance	10
Assignments	5
Quizzes	15
Lab	20
Midterm Exam	20
Final Exam	30

I. CO Assessment Plan:

Assessment Tools	**Course Outcomes**							
	CO1	**CO2**	**CO3**	**CO4**	**CO5**	**CO6**	**CO7**	**CO8**

Quizzes	√	√	√	√	√	√		
Assignments	√	√	√	√	√	√		
Lab							√	√
Midterm exam	√	√	√	√				
Final Exam	√		√	√	√	√		

Introduction to Robotics

A. Course General Information:

Course Code:	CSE 461
Course Title:	Introduction to Robotics
Credit Hours (Theory + Lab):	3+0
Contact Hours (Theory + Lab):	3+3
Category:	Program Core
Type:	Required, Engineering, Lecture
Prerequisites:	CSE 260, CSE 341, CSE360
Co-requisites:	None

B. Course Catalog Description (Content):

Design using MSI and LSI components. Design of memory subsystems using SRAM and DRAM. Design of various components of a computer: ALU, memory and control unit: hardwired and micro programmed. Microprocessor based designs. Computer bus standards. Design using special purpose controllers, floppy disk controllers. Digital control system. Computers in telecommunication and control. The course includes a compulsory 3 hour laboratory work each week.

C. Course Objective:

This course aims to introduce the exciting multi-disciplinary field of robotics to CSE students. We aim to guide the students through the electrical, mechanical, and software aspects of robotics to help them understand the basic principles and the science behind robotics and teach them how to design and develop the hardware and software parts of a basic robot. The objective of CSE461 is to develop the knowledge necessary to understand the theory behind the design of robots and provide hands-on practice on implementing that knowledge. Our goal is to teach the students the following topics that will help them understand robotics and pave the way for future research in advanced robotics.

The main objectives of this course are:

a) Introduce basic laws, architectures, control systems, perceiving techniques, and communication.
b) Introduce kinematics, motor action, mechanical engineering issues, advanced communication protocols, and uses of AI in robotics.

c) Teach how to implement different algorithms to 4DOF arm control, including forward and reverse kinematics, robot control system, and navigation.
d) Introduce camera vision, sensors, machine learning, deep learning, and other AI algorithms related to robot vision.
e) Introduce robot communication protocols including I2C, UART, USB, SPI, RS 485, Canbus, Modbus, LoRa, Bluetooth, RF, Zigbee, Wifi, and other modern protocols.
f) Teach how to design, develop, experiment, and document a robotics project.

D. Course Outcomes (COs):

Upon completing this course, students will be able to

Sl.	CO Description	Weightage (%)
CO1	**Understand** basic robotics that includes the law of robotics, uses of a robot, mechanical aspect of robot, type of primitive architecture, perceiving the environment, motor action, and different types of processing.	25
CO2	**Implement** different algorithms to 4DOF arm control, including forward and reverse kinematics. Examine Robot Control system and Navigation	20
CO3	**Relate** camera vision, sensors, machine learning, deep learning, and other AI algorithms with robot vision.	20
CO4	**Categorize** robot communication protocols (I2C, UART, USB, SPI, RS 485, Canbus, Modbus, LoRa, Bluetooth, RF, Zigbee, Wifi)	15
CO5	**Investigating** different types of robots, their characteristics, features, and applications, Design and Develop a Robot to solve a real-life problem.	20

Comprehension Application Analysis Synthesis Evaluation

E. Mapping of CO-PO-Taxonomy Domain & Level- Delivery-Assessment Tool:

Sl.	CO Description	POs	Bloom's taxonomy domain/level	Delivery methods and activities	Assessment Tools
CO1	**Understand** basic robotics that include the law of robotics, uses of robot, mechanical aspect of robot, type of primitive architecture, perceiving the environment, motor action and different types of processing.	PO1	Cognitive/ Comprehension	Lectures, notes	Quiz, Exam Homework
CO2	**Implement** different algorithms to 4DOF arm control including forward and reverse kinematics. Examine Robot Control system and Navigation	PO3	Cognitive/Apply	Lectures, notes	Quiz, Exam, Homework

CO3	**Relate** camera vision, sensors, machine learning, deep learning and other AI algorithms with robot vision.	PO5	Cognitive/ Comprehension	Lectures, notes	Quiz, Exam
CO4	**Categorize** robot communication protocols (I2C, UART, USB, SPI, RS 485, Canbus, Modbus, LoRa, Bluetooth, RF, Zigbee, Wifi)	PO2	Cognitive/Analyze	Lectures, notes	Quiz, Exam
CO5	**Investigating** different types of robots, their characteristics, features and the applications, Design and Develop a Robot to solve a real life problem.	PO12	Cognitive/Create	Lectures, notes, lab work	Exam, Homework, Lab, Presentation

F. Course Materials:

i. Text and Reference Books:

Sl.	Title	Author(s)	Publication Year	Edition	Publisher	ISBN
1	Introduction to AI robotics	Murphy, Robin	2007	1st Edition	Morgan Kaufmann	978-0262133838
2	Intelligent systems and robotics	George W. Zobrist and C.Y. Ho.	2000	1st Edition	CRC Press	978-0367398859
3	Springer Handbook of Robotics	Bruno Siciliano, Oussama Khatib	2016	2nd Edition	Springer, Cham	978-3-319-32550-7
4	Robotics, Vision and Control: Fundamental Algorithms In MATLAB, Second Edition	Peter Corke	2017	2nd Edition	Springer	978-3319544120
5	Modern Robotics: Mechanics, Planning, and Control	Kevin M. Lynch	2017	1st Edition	Cambridge University Press	978-1107156302
6	Probabilistic Robotics	Sebastian Thrun, Wolfram Burgard, Dieter Fox	2005	1st Edition	The MIT Press	978-0262201629
7	Introduction to Robotics	S K Saha	2010	1st Edition	Tata McGraw-Hill Education	978-0070140011

8	Robotics: Control, Sensing, Vision, and Intelligence	K S Fu, Rafael C. Gonzalez, C S G Lee	2010	1st Edition	Tata McGraw-Hill Education	978-0071004213

ii. Other materials (if any)

Lecture Notes and other material will be made available on buX.

G. Lesson Plan:

No	Topic	Week/Lecture#	Related CO (if any)
1	Introduce basic robotics that include the law of robotics, uses of robot, mechanical aspect of robot, type of primitive architecture, perceiving the environment, motor action, mechanical design, different types of processing and recent robotic trends.	1, 2, 3	C01
2	Review on Linear algebra and trigonometry, Robot Arm Forward Kinematics, Robot Arm Inverse Kinematics	4, 5	CO2
3	Robot Vision and Perception including vision sensors, visual servoing, physical sensors and LIDAR.	5, 6	CO3
Midterm			
4	Control Theory: Classic Feedback Diagram, First-Order and Second-Order Systems, PID Controller Navigation: Basics of Navigation, Localization techniques and Mapping	7,8	CO3
5	Applications of AI and Machine Learning in Computer	9	CO2
6	Introduce robot communication protocols including I2C, UART, USB, SPI, RS 485, Canbus, Modbus, LoRa, Bluetooth, RF, Zigbee, Wifi and other modern protocols.	10, 11	CO4
7	Case Study and Presentation.	12	CO5
Final			

H. Assessment Tools:

Assessment Tools	Weightage (%)

1.	Participation in class	5%
2.	Quizzes/Class Tests	15%
3.	Assignments	10%
4.	Mid Term Examination	20%
5.	Final Examination	30%
6.	Lab	20%
Total		100%

I. CO Assessment Plan:

Assessment Tools	**Course Outcomes**				
	CO1	**CO2**	**CO3**	**CO4**	**CO5**
1. Quizzes/Class Tests	√	√	√	√	√
2. Mid-Term Examination	√	√	√		
3. Assignments	√	√			√
4. Final Exam	√	√	√	√	√

Software Engineering

A. Course General Information:

Course Code:	CSE 470
Course Title:	Software Engineering
Credit Hours (Theory):	3
Contact Hours (Theory):	3
Category:	core
Type:	Lecture

B. Course Catalog Description (Content):

Concepts of software engineering: requirements definition, modular, structure design, data specifications, functional specifications, verification, documentation, software maintenance, Software support tools. Software project organization, quality assurance, management and communication skills.

C. Course Objective

a. Introduce Students with fundamental concepts of SDLC
b. Teach students about requirement analysis of the system
c. Teach students about how software engineers work in industry and the rules they follow.
d. Explain how to build softwares using different design patterns.
e. Teach students about managing programming process and documentation
f. Help students to build a system using proper and modern tools
g. Explain different software building processes and methodology.

D. Course Outcomes (COs):

Upon successful completion of this course, students will be able to

Sl.	CO Description	Weightage (%)
CO1	**Explain** the fundamental concepts of Software Engineering	5
CO2	**Analyze** requirements and high level overview of the software	15
CO3	**Apply** different software architecture and patterns including MVC	25
CO4	**Exhibit** an understanding of version controlling and apply it	10
CO5	**Evaluate** the viability of a software using testing and software metrics	20
CO6	**Design** a software project using development tools.	25

E. Mapping of CO-PO-Taxonomy Domain & Level- Delivery-Assessment Tool:

Sl.	CO Description	POs	Bloom's taxonomy domain/level	Delivery methods and activities	Assessment tools
CO1	Explain the fundamental concepts of Software Engineering.	PO1	Cognitive/Analyze	Lectures, notes	Quiz,exam
CO2	Analyze requirements and high level overview of the software	PO2	Cognitive/Analyze	Lectures, notes	Quiz,exam, Project
CO3	Apply different software architecture and patterns including MVC	PO3	Cognitive/Create	Lectures, notes, Project	Quiz,exam, Project
CO4	Exhibit an understanding of version controlling and apply it	PO3	Cognitive/Analyze	Lectures, notes, Project	Exam, Project
CO5	Evaluate the viability of a software using testing and software metrics	PO4	Cognitive/Analyze	Lectures, notes, Project	Quiz,exam, Project
CO6	Design a software project using development tools.	PO11	Cognitive/Analyze	Project	Project

F. Course Materials:

i. Text and Reference Books:

S	Tit	Author(s)	Publicatio	Editio	Publis	ISB

l .	le		n Year	n	her	N
1	Clean Code: A Handbook of Agile Software Craftsmanship	Robert C. Martin	2017	–	PHI	10: 9780132350884
2	Design Patterns: Elements of Reusable Object-Oriented Software	Erich Gamma, Richard Helm, Ralph Johnson and John Vlissides	1994	–	Addison Wesley	10: 0201633612

ii. **Other materials (if any)**

Lecture Notes and presentation slides

iii. Lesson Plan:

No	Topic	Week/Lecture#	Related CO (if any)
1	Introduction to software engineering and software development lifecycle.	Week 1	CO1
2	Identifying the process model based on business value,requirement of a project, preparing project team and feasibility analysis	Week 2, 3	CO2
3	Requirement engineering,functional and nonfunctional requirements	Week 4	CO3
4	Use case,Class diagrams. Notations of these diagrams, how to draw these diagrams from requirements,architectural design pattern introduction	Week 5	CO3
Mid Exam			
5	Different software architecture and patterns including MVC	Week 7,8	CO4
5	Getting familiar with version controlling and apply version controlling on the project,introduction to other agile project management tools	Week 8	CO5
6	Introduction to Software Quality Assurance and maintenance	Week 9	CO6
7	Unit testing and test driven development, Code refactoring	Week 10,11	CO6
8	Calculation techniques of different software quality and life cycle metrics	Week 11,12	CO6
Final Exam			

G. Assessment Tools:

Assessment Tools	Weightage (%)
1. Participation in class	5 %
2. Quizzes/Class Tests/Assignments/	25 %
3. Mid Term Examination	25 %
4. Project	15 %
5. Final	30 %

H. CO Assessment Plan:

Assessment Tools	CO1	CO2	CO3	CO4	CO5	CO6
Homework	x	x		x	x	
Quizzes	x	x		x	x	
Examinations	x	x	x	x	x	x
Project	x	x	x	x	x	x

System Analysis and Design

A. Course General Information:

Course Code:	CSE 471
Course Title:	System Analysis and Design System Analysis and design Laboratory
Credit Hours (Theory + Lab):	3+0
Contact Hours (Theory + Lab):	3+3
Category:	Core
Type:	Lecture + laboratory

B. Course Catalog Description (Content):

Introduces students to tools and techniques in systems analysis and design such as data flow diagram and E-R diagrams. Projects by students where they analyze the requirements and design a system using these tools. The course includes a compulsory 3-hour laboratory work each week.

C. Course Objective

a. Introduce Students with fundamental concepts of System analysis and design

b. Teach students about structural model of the system

c. Teach behavioral models of the system.

d. Explain how to make a physical model of the system.

e. Teach students about managing programming processes and documentation

f. Help students to build a system using proper and modern tools

D. Course Outcomes (COs):

Upon successful completion of this course, students will be able to

Sl.	CO Description	Weightage (%)
CO 1	**Explain** the fundamental concepts of System analysis and design.	15
CO 2	**Design** and **Analyze** different software model of the system i.e. structural, behavioral, physical	60
CO 3	**Display** teamwork through active participation in group project-related activities i.e. preparing report and oral presentation	10
CO 4	**Select** and **apply** modern software design tools to develop a project.	15

E. Mapping of CO-PO-Taxonomy Domain & Level- Delivery-Assessment Tool:

Sl.	CO Description	POs	Bloom's taxonomy domain/level	Delivery methods and activities	Assessment tools
CO1	**Explain** the fundamental concepts of System analysis and design.	a	Cognitive/Analyze	Lectures,notes	Quiz,exam
CO2	**Design** and **Analyze** different software model of the system i.e. structural, behavioral, physical	c	Cognitive/Evaluate	Lectures,notes	Assignment, Quiz,exam
CO3	**Display** teamwork through active participation in group project-related activities i.e. preparing report and oral presentation	i	Affective/Characterization	Discussion, Group Work	Lab work
CO4	**Select** and **apply** modern software design tools to develop a project.	e	Psychomotor/Affective, Cognitive/Create	Lab class	Lab work, Design Project

F. Course Materials:

i. Text and Reference Books:

Sl.	Title	Author(s)	Publication Year	Edition	Publisher	ISBN
1	Systems Analysis and Design	Dennis, Wixom and Roth	2012	5th	Prentice Hall	10: 0130415715
2	Systems Analysis and Design Methods	Jeffrey Whitten and Lonnie Bentley	2005	7th	McGraw Hill/Irwin	10: 0073052337

ii. Other materials (if any)

Lecture Notes and presentation slides

Lab Handouts

G. **Lesson Plan:**

No	Topic	Week/Lecture#	Related CO (if any)
1	Introduction to elements of system analysis and design, steps of software development life cycle.	**Week 1**	CO1
2	Identifying the business value of the new project, preparing system requests, analyze three types of feasibility study	**Week 2, 3**	CO1
3	Major components of behavioral UML diagrams such as Use case, Activity, and sequence. Notations of these diagrams, how to draw these diagrams from requirements.	**Week 4, 5, 6**	CO2
Mid Exam week			
4	Major components of structural UML diagrams such as Data flow diagrams. Notations of these diagrams, how to draw these diagrams from requirements.	**Week 8, 9**	CO2
5	Convert logical to physical process models, Designing programs, Structure charts and Program specifications.	**Week 10, 11**	CO2
6	Moving from logical to physical data models, Different data storage formats and Optimizing data storage.	**Week 12**	CO2

7	Principles of user interface design, User interface design process, Navigation design, Input design and Output design.	**Week 13**	CO2
Final Exam			

(** Please insert separate table of lesson plan for lab component if there is any)

H. Lab plan

No	Topic	Week/Lecture#	Related CO (if any)
1	Introduction to elements of system analysis and design	**Week 1**	
2	Group formation, tutorial on Trello and discuss different aspect of building a software system	**Week 2**	
3	Project Idea finalization and preparing system request	**Week 3**	
4	Analyze functional and non-functional requirements	**Week 4**	
5	Prepare SRS and Design Use case diagram	**Week 5**	
6	Design activity diagram	**Week 6**	
7	Design sequence diagram	**Week 7**	
8	Design Dataflow diagram	**Week 8**	
9	Design windows navigation diagram and project implementation	**Week 9**	
10	Designing system using suitable tools	**Week 10**	CO4
11	Documentation preparation, Report writing and preparing presentation	**Week 11**	CO4
12	Final presentation and demonstration	**Week 12**	CO4

I. Assessment Tools:

Assessment Tools	Weightage (%)
1. Participation in class	5 %
2. Quizzes/Class Tests/Assignments/	10+10 %
3. Mid Term Examination	25 %
4. Lab & Project	25 %
5. Final	25 %

J. CO Assessment Plan:

Assessment Tools				
	CO1	CO2	CO3	CO4
Homework		x		
Quizzes	x	x		
Examinations	x	x		
Laboratory Works			x	
Project				x

Final Year Design Project

A. Course General Information:

Course Code:	CSE400
Course Title:	Final Year Design Project
Credit Hours	4
Contact Hours	3 Hours/week/semester - I Year (3 Semester)
Category:	School/Program Core
Type:	Required, Engineering
Prerequisites:	CSE361 Digital Interfacing and System Design CSE369 Professional Practice, Engineers and Society CSE421 Computer Networks CSE422 Artificial Intelligence CSE470 Software Engineering
Credit requirements:	Minimum 100 credit hours completed

B. Course Catalog Description (Content):

The Final Year Design Project is the first step towards transferring students' experience from the academic environment to the industry. The course provides a culminating assessment of the students by applying and integrating their previously acquired knowledge to the solution of complex engineering and computing problems. The primary focus of the Final Year Design Project is to improve the students' technical skills, communication skills and teamwork opportunities through engineering project development work. It also focuses on a variety of non-technical issues such as professional and ethical responsibilities and practices, safety, reliability, legal cultural, social and environmental impacts as well as sustainability of engineering solutions.

The Final Year Design Project course consists of two parts: Instructional Part and Demonstration Part.

The major topics covered in the Instructional Part include:

- Overview of the Final Year Design Project course, student learning outcomes, expectation, assessment, checklist etc.
- Introduction to engineering design process including formulation of problem, analysis of objectives, specifications and requirements, consideration of realistic constraints, engineering standards and impact of engineering solutions, design of solution, implementation, evaluation and validation of the solution
- Review of project proposal preparation, estimating, project management and scheduling etc.
- Review of engineering ethics and professional practices
- Safety in engineering design.
- Contemporary issues and life-long learning
- Report writing and presentation techniques
- Teamwork building

The Demonstration part primarily includes various activities including (but not limited to):

- Literature review and research
- Identification and formulation of project problem
- Analysis of objectives, specifications and requirements,
- Project plan, proposal and management
- Implementation of design process
- Design reviews, simulation and finalization
- Development of solution, testing and validation
- Documentation, drawings, written reports, oral presentation etc.

All these activities of the Demonstration part are conducted under three phases- Phase 1: Problem Identification and Project Planning, Phase 2: Design Solutions and Phase 3: Development and Validation.

C. Course Objectives:

The objectives of the Final Year Design Project are to:

a. Provide students opportunity to apply and integrate their previously acquired engineering knowledge to the solution of engineering problem

b. Enhance student’s creativity in analyzing and solving complex and possibly real-world engineering problems.
c. Train students with skills on systematic design and development process and documentation to the solution of engineering project
d. Prepare students to develop and enhance self-learning ability.
e. Prepare students experience of engineering project development that will be useful in their industrial careers.
f. Aware students regarding professional practices, norms and ethical responsibilities in regards to designing engineering solution
g. Prepare student to understand and evaluate the impact of engineering solutions to the society, health, safety, reliability, legal, cultural social
h. Prepare students to understand and evaluate the sustainability and impact of engineering solution towards environment
i. Create an environment to promote team approach in engineering problem solving
j. Develop communication skills among students through complex activities, technical report writing, oral presentations etc.

D. Course Outcomes (COs):

Upon successful completion of this course, students will be able to

Sl.	CO Definition	Weightage
CO1	Formulate a modern-day complex engineering and/or computing problem preferably relevant to the industry	5%
CO2	Identify the objectives, specifications, functional and non-functional requirements, and constraints as well as applicable compliance, standards and codes of practice to the solution of the engineering problem	5%
CO3	Assess the impact of the solution of the engineering project in terms of societal, health, safety, legal and cultural context	5%
CO4	Evaluate the sustainability and impact of solution of the proposed project in terms of environmental consideration	5%
CO5	Design multiple engineering solutions of the problem to meet the desired objectives, need and requirements within the given constraints	10%
CO6	Analyze alternative design solutions of engineering problems in order to find the most appropriate one considering cost, efficiency, usability, manufacturability, impact, sustainability, maintainability etc.	10%

CO7	Validate the performance of the developed solution with respect to the given specifications, requirements and standards	5%
CO8	Complete the final design and development of the solution with necessary adjustment based on performance evaluation	5%
CO9	Use modern engineering and IT tools to design , develop and validate the solution	10%
CO10	Conduct independent research, literature survey and learning of new technologies and concepts as appropriate to identify, analyze, and formulate the problem as well as design, develop and validate the solution	5%
CO11	Demonstrate project management skill in various stages of developing the solution of engineering design project	10%
CO12	Perform cost-benefit and economic analysis of the solution	5%
CO13	Apply ethical considerations and professional responsibilities in designing the solution and throughout the project development phases	5%
CO14	Function effectively as an individual and as a team member for successfully completion of the project	5%
CO15	Communicate effectively through writings, journals, technical reports, deliverables, presentations and verbal communication as appropriate at various stages of project development	10%