NRI INSTITUTE OF TECHNOLOGY

(An Autonomous Institution Permanently Affiliated to JNTUK, Kakinada)
(Accredited by NAAC with "A" Grade and ISO 9001:2015 Certified Institution)
POTHAVARAPPADU (V), (VIA) NUNNA, AGIRIPALLI (M), PIN – 521 212

DEPARTMENT OF INFORMATION TECHNOLOGY
COURSE STRUCTURE FOR SECOND YEAR B.TECH PROGRAMME

II YEAR I SEMESTER

Sl. No	Course Code	Title of the Course	Scheme of Instruction (Periods Per Week)				Scheme of Examination (Maximum Marks)			No. of Credits
			L	T	P/D	Total	CIA	SEA	Total	
1	18A2100202	Discrete Mathematical Structures	3	1	0	4	40	60	100	4
2	18A2105401	Data Structures	3	0	0	3	40	60	100	3
3	18A2105402	Data Base Management Systems	3	0	0	3	40	60	100	3
4	18A2105403	Digital Logic Design	2	0	2*	4	40	60	100	3
5	18A2105493	Internet of Things Lab	0	2	2	4	40	60	100	3
6	18A2105491	Data Structures Lab	0	0	2	2	40	60	100	1
7	18A2105492	Data Base Management System Lab	0	0	2	2	40	60	100	1
8	18A2105494	Python programming Lab	0	1	2	3	40	60	100	2
9	HSS Elective	Humanities elective-1	2	0	0	2	40	60	100	1
10	18A2100802	Professional ethics and Human Values	2	0	0	2	40	60*	100	0
Total			**15**	**4**	**10**					**21**

* No External Evaluation
List of Humanities Electives

A	Professional Communication Skills(18A2100601)	D	Psychology (18A2100604)
B	Visual Communication(18A2100602)	E	Philosophy (18A2100605)
C	Sanskrit (18A2100603)	F	Foreign Languages (German/French) (18A2100606)

**Students can opt any one of the self-learning courses prescribed by the Department. Students register and complete the opted course in approved MOOCS platform on or before the Last Instruction Day of the current semester. They have to submit the certificate before the commencement of the next semester.

NRI INSTITUTE OF TECHNOLOGY

(An Autonomous Institution Permanently Affiliated to JNTUK, Kakinada)
(Accredited by NAAC with "A" Grade and ISO 9001:2015 Certified Institution)
POTHAVARAPPADU (V), (VIA) NUNNA, AGIRIPALLI (M), PIN – 521 212

II YEAR II SEMESTER

S.No	Course Code	Title of the Course	Scheme of Instruction (Periods Per Week)				Scheme of Examination (Maximum Marks)			
			L	T	P	Total	CIA	SEA	T	Credits
1	18A2200201	Probability and Statistics	3	1	0	4	40	60	100	4
4	18A2205401	Web Technologies and Advanced Java Programming	3	0	0	3	40	60	100	3
3	18A2205402	Software Engineering	3	0	2*	5	40	60	100	4
2	18A2205403	Computer Organization	3	0	0	3	40	60	100	3
5	18A2212601	**Open Elective-1 Software Engineering**	3	0	0	3	40	60	100	3
6	18A2205491	Web Technologies and Advanced Java Programming Lab	0	0	3	3	40	60	100	1.5
7	18A2205991	Aptitude and Reasoning	0	0	2	2	40	60*	100	1
8	18A2205791	Mini project	0	0	2	2	40	60*	100	1
Total			15	1	9					20.5

* No External Evaluation

L - LECTURE   T – TUTORIAL   P - PRACTICAL
CIA – Continuous Internal Assessment   SEA – Semester End Assessment

B.TECH IT
II YEAR-I SEMESTER

18A2100203- DISCRETE MATHEMATICAL STRUCTURES

Lecture – Tutorial-Practical:	3-1-0	Internal Marks:	40
Credits:	4	External Marks:	60

Prerequisites:

Course Objectives:

Acquiring the relevance of statements, inferences and predicates in computer science.
Overview of group theory.
Overview of recurrence relations and solving recurrence relations.
Exposure of graphs, their representations, types, trees and tree variants.

Course Outcomes:

Upon successful completion of the course, the student will be able to:

CO1	Apply the concept of Mathematical Logic in software development process.
CO2	Apply the recurrence relation for analyzing recursive algorithms.
CO3	Student will able to understand the concepts of group theory.
CO4	Apply the concept of group theory in robotics, computer vision & computer graphics.
CO5	Student will able to understand the concepts of graph theory and Trees.
CO6	Use the concepts of graph theory to provide solutions for routing applications in computer networks.

Contribution of Course Outcomes towards achievement of Program Outcomes (1 – Low, 2- Medium, 3 – High)

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO1	3	3	3	-	-	-	-	-	-	-	-	-
CO2	3	3	2	-	-	-	-	-	-	-	-	-
CO3	2	2	2	-	-	-	-	-	-	-	-	-
CO4	3	2	2	-	-	-	-	-	-	-	-	-
CO5	2	2	2	-	-	-	-	-	-	-	-	
CO6	3	3	3	-	-	-	-	-	-	-	-	

UNIT I :Mathematical logic

Connectives, negation, conjunction, disjunction, conditional and bi-conditional, well formed formulae, tautologies, equivalence of formulae, duality, tautological implications, functionally complete set of connectives, principal disjunctive and conjunctive normal forms, inference calculus, rules of inference, indirect method of proof, conditional proof, automatic theorem proving.

UNIT II: Recurrence relations

Recurrence relations, solving linear recurrence relations by characteristic roots method, system of recurrence relations, non - linear recurrence relations.

UNIT III: Group theory

Groups, subgroups, Lagrange's theorem on finite groups, normal subgroups. group codes

UNIT IV: Graph theory & Trees

Definitions, finite and infinite graphs, incidence and degree, isolated pendant vertices,

isomorphism, sub graphs, walk, path and circuit, connected and disconnected graphs, components, Euler graphs, Euler graph theorem, operations on graphs, decomposition of Euler graphs into circuits, arbitrarily traceable Euler graphs, Hamiltonian paths and circuits, number of edge disjoint Hamiltonian circuits in complete graph with odd number of vertices, travelling salesman problem. Some properties of trees, pendant vertices, distance and centers, rooted and binary trees, spanning trees, fundamental circuit, shortest spanning trees, Kruskal's algorithm.

TEXT BOOKS:

1. J.P. Tremblay and R. Manohar, Discrete Mathematical Structures with Applications to Computer   Science, Tata McGraw Hill, 1997. (Modules 1 and 3)
2. Joe L. Mott, Abraham Kandel and T. P. Baker, Discrete Mathematics for computer
3. scientists & Mathematicians, 2/e, Prentice Hall of India Ltd, 2012. (Module 2)
4. Narsingh Deo, Graph Theory with Applications to Engineering and Computer Science, Prentice Hall of India, 2006. (Modules 4 and 5).

REFERENCE BOOKS:

1. Keneth. H. Rosen, Discrete Mathematics and its Applications, 6/e, Tata McGraw-Hill, 2009.
2. Richard Johnsonburg, Discrete mathematics, 7/e, Pearson Education, 2008

E-RESOURCES:

18A2105401- DATA STRUCTURES

Lecture – Tutorial-Practical::	3-0-0	Internal Marks:	40
Credits:	3	External Marks:	60
Prerequisites:			

C- Programming

Course Objectives:

1 To impart basic knowledge of data structures.

2 Be familiar with basic techniques of algorithm analysis

3 Be familiar with writing recursive methods

4 To understand concepts about searching and sorting techniques

5 To design and implementation of various basic and advanced data structures like stacks, queues, lists, trees and graphs.

6 To introduce various techniques for representation of the data in the real world.

To understanding about writing algorithms and step by step approach in solving problems with the help of fundamental data structures

Course Outcomes:

Upon successful completion of the course, the student will be able to:

CO1	Ability to illustrate the concepts of algorithm apply the learning concepts to design data structure for the given problem definition.
CO2	Ability to design applications using stacks and queues and implements various types of queues
CO3	Analyze and implement operations on linked list and demonstrate their applications
CO4	Ability to analyze and implement operations on trees
CO5	Ability to demonstrate various operations on binary search trees and its applications
CO6	Ability to evaluate the properties and operations on graphs and implement the graph applications

Contribution of Course Outcomes towards achievement of Program Outcomes (1 – Low, 2-Medium, 3 – High)

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO1	3	3	2	2	-	-	-	-	-	-	-	3
CO2	3	2	3	2	-	-	-	-	-	-	-	3
CO3	3	2	3	2	-	-	-	-	-	-	-	3
CO4	3	2	3	2	-	-	-	-	-	-	-	3
CO5	3	2	3	2	-	-	-	-	-	-	-	3
CO6	3	2	3	2	-	-	-	-	-	-	-	3

UNIT I : Data Structures, Recursion, Searching, and Sorting.

Data Structures: Definition, Types of Data Structures, Arrays, structures, self-referential structures, Operations, Algorithm analysis Time Complexity and Space Complexity.

Recursion: Definition, Linear and Binary recursions, Iteration vs. Recursion.

Searching: Linear Search, Binary Search.

Sorting: Basic concepts, Divide-and-Conquer approach, Insertion Sort, Merge Sort, Quick Sort, and Heap Sort.

UNIT II: Linked Lists, Stacks, and Queues.

Linked Lists: Introduction, types of Linked Lists, operations, inserting a node in Single Linked List, deleting a node in Single Linked List, searching a node in Single Linked List, inserting, deleting, and searching a node in Double Linked List.

Stacks: Introduction, operations, applications, Stacks implementation using Arrays, Stacks implementation using Linked List, Expression Conversion: Infix to Postfix, Infix to Prefix.

Queues: Introduction, operations, applications, Queues implementation using Arrays, Queues implementation using Linked Lists, Circular Queue. Priority Queues

UNIT III: Trees.

Basic Tree Concepts, Terminology, operations, Tree traversals, **Binary Trees:** definition, properties, Binary Tree representations, operations, **Binary Search Tree:** definition, properties, applications, Inserting, Deleting, and Searching element in Binary Search Tree,

Threaded Binary Tree: definition, properties, Inserting a Node into a Threaded Binary Tree, **Heaps:** Definition of a Max Heap, properties.

UNIT IV: Graphs.

Graphs: Introduction, Terminology, Representation of graphs, types of graphs, applications, operations, Graph transversal techniques: Breadth First Search (BFS), Depth First Search (DFS), implementations. **Minimum Spanning Tree (MST):** definition, Prim's algorithm, Kruskal's algorithm, **Shortest paths:** Basic Concepts, Dijsktra's algorithm

TEXT BOOKS:

1. Fundamentals of DATA STRUCTURES in C, Horowitz,Sartaj Sahani, Susan Anderson – Freed, University Press

2.Data Structures, 2/e, Richard F, Gilberg , Forouzan, Cengage

REFERENCE BOOKS:

1 .Data Structures using C, 2nd Edition,by A. K. Sharma, Pearson India

2. Classic Data Structures, 2/e, Debasis, Samanta, PHI,2009

3. Data Structures and Algorithms, 2008, G.A.V.Pai, TMH

4. DATA STRUCTURE USING C,Udit Agarwal,KATSON Books

5. Data Structures using C,Reema Thareja, Oxford

E-RESOURCES:

1. https://en.wikipedia.org/wiki/Data_structure
2. https://www.tutorialspoint.com/data_structures_algorithms/data_structures_basics
3. http://nptel.ac.in/courses/106103069/

18A2105402- DATA BASE MANAGEMNET SYSTEMS

Lecture – Tutorial- Practical::	3-0-0	Internal Marks:	40
Credits:	3	External Marks:	60

Prerequisites:	

C- Programming, Mathematics

Course Objectives:
- Learn and practice data modeling using the entity-relationship and developing database designs
- Understand the use of Structured Query Language (SQL) and learn SQL syntax.
- Apply normalization techniques to normalize the database
- Understand the needs of database processing and learn techniques for controlling the consequences of concurrent data access
- Understand the indexing concepts and how help full in accessing data
- Learn the concepts of transaction management and how they provide security and consistency

Course Outcomes:

Upon successful completion of the course, the student will be able to:

CO1	Ability to define, understand the database management system structure
CO2	Ability to apply SQL as well as relational algebra to find solutions to a broad range of queries.
CO3	Ability to design ER diagrams or database for given scenario
CO4	Ability to create applications using various normal forms, functional dependencies, validating and identifying anomalies.
CO5	Ability to design application using locking methods and recovery management.
CO6	Ability to conduct experiments of database using modern tools.

Contribution of Course Outcomes towards achievement of Program Outcomes (1 – Low, 2- Medium, 3 – High)

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO1	-	-	-	-	-	-	-	-	-	-	-	-
CO2	3	-	-	-	-	-	-	-	-	-	-	-
CO3	-	3	-	-	-	-	-	-	-	-	-	-
CO4	-	-	3	-	-	-	-	-	-	3	-	-
CO5	-	-	-	3		-	-	-	-	-	-	-
CO6	-	-		-	3	-	-	-	-	3	-	-

UNIT I

Databases and Database Users: Introduction, Characteristics of the Database Approach, Actors on the Scene, Workers behind the Scene, Advantages of Using the DBMS Approach.

Database System Concepts and Architecture: Data Models, Schemas, and Instances, Three-Schema Architecture and Data Independence, Database Languages and Interfaces, The Database System Environment, Centralized and Client/Server

Architectures for DBMSs
UNIT II
SQL: SQL Data Definition and Data Types, Specifying Constraints in SQL, Schema Change Statements in SQL , INSERT, DELETE, and UPDATE Statements in SQL, Basic Retrieval Queries in SQL, More Complex SQL Retrieval Queries, Views (Virtual Tables) in SQL.
The Relational Algebra: Unary Relational Operations: SELECT and PROJECT, Relational Algebra Operations from Set Theory, Binary Relational Operations: JOIN and DIVISION.
Data Modeling Using the Entity-Relationship (ER) Model: Using High-Level Conceptual Data Models for Database Design, A Sample Database Application, Entity Types, Entity Sets, Attributes, and Keys, Relationship Types, Relationship Sets, Roles, and Structural Constraints, Weak Entity Types, Relational Database Design Using ER-to-Relational Mapping.
UNIT III
Normalization: Functional Dependencies, Inference Rules, Equivalence, and Minimal Cover, Properties of Relational Decompositions, Algorithms for Relational Database Schema Design, Normal Forms Based on Primary Keys, General Definitions of Second and Third Normal Forms, Boyce-Codd Normal Form, Multivalued Dependency and Fourth Normal Form, Join Dependencies and Fifth Normal Form.
File Structures, Hashing and Indexing: Placing File Records on Disk, Operations on Files, Files of Unordered Records (Heap Files), Files of Ordered Records (Sorted Files), Hashing Techniques, Types of Single-Level Ordered Indexes, Multilevel Indexes, Dynamic Multilevel Indexes Using B-Trees and B+-Trees.
UNIT IV:
Transaction Processing: Transaction and System Concepts, Desirable Properties of Transactions, Characterizing Schedules Based on Recoverability, Characterizing Schedules Based on Serializability.
Concurrency Control Techniques: Two-Phase Locking Techniques for Concurrency Control, Concurrency Control Based on Timestamp Ordering, Validation (Optimistic) Concurrency Control Techniques
Database Recovery Techniques: Recovery Concepts, NO-UNDO/REDO Recovery Based on Deferred Update, Recovery Techniques Based on Immediate Update, Shadow Paging, The ARIES Recovery Algorithm
TEXT BOOKS:
1. Fundamentals of Database Systems, Elmasri Navrate, 6th edition, Pearson Education
REFERENCE BOOKS:
1. "Database System Concepts", . Korth, Slberchatz, Sudarshan, 6th Edition, McGraw – Hill 2. Peter Rob and Carlos Coronel, " Database Systems Design, Implementation and Management", Thomson Learning, 5th Edition. 3. Introduction to Database Systems, CJ Date, Pearson 4. DATA STRUCTURE USING C,Udit Agarwal,KATSON Books 4. Data base Management Systems, Raghurama Krishnan, Johannes Gehrke, TATA McGraw Hill 3rd Edition
E-RESOURCES:

18A2105403- DIGITAL LOGIC DESIGN

Lecture – Tutorial- Practical:	2-0-2	Internal Marks:	40
Credits:	3	External Marks:	60
Prerequisites:			

Number Systems, Mathematics-I , Mathematics-II

Course Objectives:

- To study the basic philosophy underlying the various number systems, negative number representation, binary arithmetic, binary codes and error detecting and correcting binary code.
- To study the theory of Boolean algebra and to study representation of switching functions using Boolean expressions and their minimization techniques.
- To study the combinational logic design of various logic and switching devices and their realization.
- To study some of the programmable logic devices and their use in realization of switching functions.
- To study the sequential logic circuits design both in synchronous and Asynchronous modes for various complex logic and switching devices, their minimization techniques and their realizations.
- To implement synchronous state machines using flip flops.

Course Outcomes:
Upon successful completion of the course, the student will be able to:

CO1	Understand the numeric information in different forms and interpret different logic gates.
CO2	Minimize the given Switching functions in SoP and PoS forms using K-Map and Tabular Method.
CO3	Analyze and Design various combinational circuits like Encoders, Decoders, Multiplexers, De-multiplexers, and Arithmetic Circuits.
CO4	Design combinational logic circuits using different types of Programmable Logic Designs.
CO5	Design and Implement various sequential circuits like flip flops, registers.
CO6	Design the state diagrams with the knowledge of Mealy and Moore conversions, state machines using various flip flops.

Contribution of Course Outcomes towards achievement of Program Outcomes (1 – Low, 2- Medium, 3 – High)

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO1	3	3	-	-	-	-	-	-	-	-	-	-
CO2	3	2	-	-	-	-	-	-	-	-	-	-
CO3	3	2	1	-	-	-	-	-	-	2	-	3
CO4	2	-	-	-	-	-	-	-	-	-	-	-
CO5	3	3	2	-	-	-	-	-	-	2	-	3
CO6	2	2	3	-	-	-	-	-	-	-	-	-

UNIT I

NUMBER SYSTEMS & BOOLEAN FUNCTIONS: Representation of numbers in different radix, conversation from one radix to another radix, r-1's compliments and r's compliments of signed numbers, weighted and non-weighted codes , Gray code, Error detection, error correction codes, parity checking, Hamming code, Boolean theorems, principle of complementation & duality, De-Morgans theorems, Basic logic gates and Universal gates, NAND-NAND and NOR-NOR realizations, Standard SOP and POS.
Practical's
1. Verification of Basic Logic Gates.
2. Implementing all individual gates with Universal Gates NAND & NOR.

Design a circuit for the given Canonical form, draw the circuit diagram and verify the De-Morgan laws

UNIT II

MINIMIZATION TECHNIQUES & COMBINATIONAL LOGIC DESIGN: Minimization techniques: minimization of logic functions using Boolean theorems, minimization of switching functions using K-Map up to 5 variables, tabular minimization, Design of Half adder, full adder half subtractor, full subtractor, 4-bit binary subtractor, adder-subtractor circuit, BCD adder circuit, Excess 3 adder circuit, 4 bit parallel adder, Carry look-a-head adder circuit, applications of adders and subtractors. Decoders, 7 segment decoder, Encoders, priority encoder, Multiplexer, Demultiplexer.

Practical's

1. Construct Half Adder and Full Adder using Half Adder and verify the truth table.
2. Design a Combinational Logic circuit for 4x1 MUX and verify the truth table.
3. Design a Combinational Logic circuit for 1x4 De- MUX and verify the truth table.

UNIT III

SEQUENTIAL LOGIC DESIGN: Classification of sequential circuits, Latches and Flipflops, Triggering, excitation tables, Asynchronous inputs, Conversion from one flip-flop to another flip flop. Registers-Types, modes of operations, bi-directional shift registers, universal shift register, Counters-synchronous & Asynchronous counters, design of Mod-counters, Counters using shift registers, Serial binary adder.

Practical's

1. Verification of truth tables of the basic Flip- Flops with *Synchronous* and *Asynchronous* modes
2. Implementation of Master Slave Flip-Flop with J-K Flip- Flop and verify the truth table for *race around* condition.
3. Design a Decade Counter and verify the truth table.
4. Design the Mod 6 counter using D-Flip –Flop.
5. Construct 4-bit ring counter with T-Flip –Flop and verify the truth table.
6. Design a 8 – bit right Shift Register using D-Flip -Flop and verify the truth table.

UNIT IV:

STATE MACHINES: Finite state machine, Analysis of clocked sequential circuits, state diagrams, state tables, reduction of state tables and state assignment, design procedures. Realization of circuits using various flip-flops. Melay to Moore conversion and vice-versa.

TEXT BOOKS:

1.Hill and Peterson "Switching Theory and Logic Design" Mc-Graw Hill TMH edition.

2.A. Anand Kumar "Switching Theory and Logic Design" PHI, 2009

EQUIPMENT REQUIRED:

Digital IC Trainer kit

REFERENCE BOOKS:

RP Jain, " Modern Digital Electronics", TMH, 2009.

Fundamentals of Logic Design by Charles H.Roth Jr, Cenage Learning, 2010

Digital Logic And Computer Design By M. Morris Mano

E-RESOURCES:

https://nptel.ac.in/courses/106108099/
https://swayam.gov.in/course/1392-digital-circuits-and-systems
http://www.nesoacademy.org/electronics-engineering/digital-electronics/digital
https://www.youtube.com/playlist?list=PLWPirh4EWFpHk70zwYoHu87uVsCC8E2S-
https://www.youtube.com/watch?v=X7M3rUxUpOc&list=PLbRMhDVUMngePP5JcezxImF-
FzOC9wstz&index=1
https://www.youtube.com/watch?v=IDf2vEcyDfs
https://www.youtube.com/watch?v=HcH0khFGwS8&list=PLbRMhDVUMngfV8C6ElNAUaQQz06wEhFM5

18A2105493- INTERNET OF THINGS LAB

Lecture – Tutorial-Practical:	0-2-2	Internal Marks:	40
Credits:	3	External Marks:	60

Prerequisites:

Number Systems, Mathematics-I , Mathematics-II

Course Objectives:
Assess the genesis and impact of IoT applications, architectures in real world
Illustrate diverse methods of deploying smart objects and connect them to network
Compare different Application protocols for IoT
Expose the student to a variety of embedded boards and IoT Platforms
Identify sensor technologies for sensing real world entities and understand the role of IoT in various domains of Industry

Course Outcomes:

Upon successful completion of the course, the student will be able to:

CO1	Interpret the impact and challenges posed by IoT networks leading to new architectural models.
CO2	Explain the basics of microcontrollers, architecture of Arduino and develop simple applications using Arduino.
CO3	Outline the architecture of Raspberry Pi and develop simple applications using Raspberry, select a platform for a particular embedded computing application
CO4	Interpret different protocols and compare them and select which protocol can be used for a specific application
CO5	Select IoT APIs for an application
CO6	Design and develop a solution for a given application using APIs and test for errors in the application

Contribution of Course Outcomes towards achievement of Program Outcomes (1 – Low, 2- Medium, 3 – High)

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO1	2	-	2	-	-	2	-	-	-	-	2	-
CO2	3	2	-	-	-	2	-	-	2	-	-	-
CO3	3	-	2	2	-	2	-	-	-	-	-	2
CO4	2	-	3	-	-	2	-	-	-	-	-	-
CO5	3	2	2	-	2	2	-	-	-	-	-	2
CO6	-	2	3	-	2	2	-	-	-	-	-	2

UNIT I

Overview of IoT:
The Internet of Things: An Overview; The Flavour of the Internet of Things, The "Internet" of "Things", The Technology of the Internet of Things, Design Principles for Connected Devices, Calm and Ambient Technology, Privacy; Keeping Secrets, Web Thinking for Connected Devices, Small Pieces, Loosely Joined, First-Class Citizens On The Internet

UNIT II

Embedded Computing Basics; Microcontrollers; System-on-Chips; Choosing Your Platform; Arduino; Developing on the Arduino; Some Notes on the Hardware; Openness;

Raspberry Pi ; Cases and Extension Boards; Developing on the Raspberry Pi; Some Notes on the Hardware; Openness; Other notable platforms; Mobile phones and tablets; Plug Computing: Always-on Internet of Things

1. Select any one development board (Eg., Arduino or Raspberry Pi) and control LED using the board.
2. Using the same board as in (1), read data from a sensor. Experiment with both analog and digital sensors.
3. Control any two actuators connected to the development board using Bluetooth.
4. Create any cloud platform account, explore IoT services and register a thing on the platform.

UNIT III

Internet Principles; Internet Communications: An Overview of IP, TCP, The IP Protocol Suite (TCP/IP), UDP, IP Addresses, Static IP Address Assignment, Dynamic IP Address Assignment, IPv6.

1. Push sensor data to cloud.
2. Control an actuator through cloud.

Accesses the data pushed from sensor to cloud and apply any data analytics or visualization services.

UNIT IV:

Getting Started with an API, Mashing Up APIs, Scraping, Legalities, Writing a New API, MQ Telemetry Transport, Extensible Messaging and Presence Protocol, Constrained Application Protocol

1. Create a mobile app to control an actuator.
2. Identify a problem in your local area or college which can be solved by integrating the things you learned so far and create a prototype to solve it (Mini Project).

TEXT BOOKS:
1. 1. **Adrian McEwen, Hakim Cassimally - Designing the Internet of Things, Wiley Publications, 2012**

REFERENCE BOOKS:
1. Vijay Madisetti and ArshdeepBahga, "Internet of Things (A Hands-on-Approach)", 1 stEdition, VPT, 2014. (ISBN: 978-8173719547)
2. **The Internet of Things, Enabling technologies and use cases – Pethuru Raj, Anupama C. Raman, CRC Press.**

E-RESOURCES:
https://www.arduino.cc/

https://www.raspberrypi.org/

18A2105491- DATA STRUCTURES LAB

Lecture – Tutorial-Practical:	0-0-2	Internal Marks:	40
Credits:	1	External Marks:	60

Prerequisites:

C Programming

Course Objectives:

- To understand and implement basic data structures
- To Apply linear and non linear data structures in problem solving.
- Have a good understanding of how several fundamental algorithms work, particularly those concerned with sorting and searching.
- Have a good understanding of the fundamental data structures used in computer science
- It enables them to gain knowledge in practical applications of data structures.

Course Outcomes:

Upon successful completion of the course, the student will be able to:

CO1	Implement different searching and sorting techniques. Compare different searching and sorting techniques.
CO2	Design linear data structures stacks, queues and linked lists.
CO3	Design nonlinear data structures trees and Graphs, and implement their operations
CO4	Be capable to identity the appropriate data structure for given problem
CO5	Have practical knowledge on the applications of data structures

Contribution of Course Outcomes towards achievement of Program Outcomes (1 – Low, 2- Medium, 3 – High)

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO1	3	3	2	2	-	-	-	-	-	-	-	2
CO2	3	2	3	2	-	-	-	-	-	-	-	2
CO3	3	2	3	2	-	-	-	-	-	-	-	2
CO4	3	2	3	2	-	-	-	-	-	-	-	2
CO5	3	2	3	2	-	-	-	-	-	-	-	2

List of Experiments

Exercise 1:
   a. Write a recursive C program to find the Factorial of an integer.
   b. Write a recursive C program to calculate the GCD of two numbers.
   c. Write a recursive C program for Towers of Hanoi: N disks are to be transferred from peg S to peg D with Peg I as the intermediate peg.
   d. Write a recursive C program to display the Fibonacci Series: 0, 1, 1, 2, 3, 5, 8, ...N.

Exercise 2:
   a. Write a recursive and non-recursive C program to implement Linear Search technique.
   b. Write a recursive and non-recursive C program to implement Binary Search

technique.

Exercise 3:
    a. Write C program that implement Insertion sort, to sort elements in an ascending order.
    b. Write C program that implement Merge sort, to sort elements in an ascending order.
    c. Write C program that implement Quick sort, to sort elements in an ascending order.

Exercise 4:
    a. Write a C program to insert a node in a Single Linked List.
    b. Write a C program to delete a node in a Single Linked List.
    c. Write a C program to reverse elements in a Single Linked List.
    d. Write a C program to insert a node in a Doubly Linked List.

Exercise 5:
    a. Write C program that implement Stack (its operations) using arrays.
    b. Write C program that implement Queue (its operations) using arrays.
    c. Write C program that implement Queue using Two Stacks.

Exercise 6:
    a. Write C program that implement Stack using Linked List.
    b. Write C program that implement Queue using Linked List.
    c. Write a C program to implement the Circular Queue.

Exercise 7:
    a. Write a C program to insert elements in a Binary Search Tree (BST).
    b. Write a C program to delete element in a Binary Search Tree (BST).
    c. Write a C program to implement BST traversals: Inorder, Preorder, and Postorder.

Exercise 8:
    a. Write a C program to implement the Max Heap.
    b. Write C program that implement Heap sort, to sort elements in an ascending order.

Exercise 9:
    a. Write a C program to implement the Breadth First Search technique on a Graph.
    b. Write a C program to implement the Depth First Search technique on a Graph.

Exercise 10:
    a. Write a C program to implement the Prim's algorithm to construct Minimum Spanning Tree.
    b. Write a C program to implement the Kruskal's algorithm to construct Minimum Spanning Tree.

TEXT BOOKS:

1 Data Structures using C,Reema Thareja, Oxford

2.DATA STRUCTURE USING C,Udit Agarwal,KATSON Books

3 Data Structures using C, 2nd Edition, by A. K. Sharma, Pearson India

REFERENCE BOOKS:

1. Keneth. H. Rosen, Discrete Mathematics and its Applications, 6/e, Tata McGraw-Hill, 2009.

2. Richard Johnsonburg, Discrete mathematics, 7/e, Pearson Education, 2008

E-RESOURCES:

18A2105492- DATABASE MANAGEMENT SYSTEMSLAB

Lecture – Tutorial-Practical:	0-2-2	Internal Marks:	40
Credits:	1	External Marks:	60

Prerequisites:

C Programming, Mathematics

Course Objectives:

- Design and implement a database schema for a given problem-domain
- Populate and query a database using SQL DML/DDL commands.
- Declare and enforce integrity constraints on a database
- Programming PL/SQL including stored procedures, stored functions, cursors, packages.
- understand real time data base design models and can code the model
- understand and retrieve information from complex designed data bases using correlated nested queries

Course Outcomes:

Upon successful completion of the course, the student will be able to:

CO1	Queries for Creating, Dropping, and Altering Tables, Views, and Constraints
CO2	Queries to Retrieve and Change Data: Select, Insert, Delete, and Update
CO3	Queries using Built-In Functions: String Functions, Numeric Functions, Date Functions and Conversion Functions.
CO4	Queries using Group By, Order By, and Having Clauses
CO5	Queries on Joins and Correlated Sub-Queries
CO6	Queries on Controlling Data: Commit, Rollback, and Save point

Contribution of Course Outcomes towards achievement of Program Outcomes (1 – Low, 2- Medium, 3 – High)

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO1	3	1	-	-	-	-	-	-	-	-	-	-
CO2	2	3	3	-	1	-	-	-	-	-	-	-
CO3	1	2	2	-	2	-	-	-	-	-	-	-
CO4	1	2	3	-	3	-	-	-	-	-	-	1
CO5	1	2	1	-	2	-	-	-	-	-	-	-
CO6	1	1	3	-	2	-	-	-	-	-	-	-

List of Experiments

SQL

1. Queries for Creating, Dropping, and Altering Tables, Views, and Constraints
2. Queries to Retrieve and Change Data: Select, Insert, Delete, and Update
3. Queries using operators in SQL
4. Queries using Built-In Functions: String Functions, Numeric Functions, Date Functions and Conversion Functions.
5. Queries using Group By, Order By, and Having Clauses
6. Queries on Controlling Data: Commit, Rollback, and Save point
7. Queries on Joins and Correlated Sub-Queries
8. Queries on Working with Index, Sequence, Synonym

PL/SQL

1. Write a PL/SQL Code using Basic Variable, Anchored Declarations, and Usage of Assignment Operation
2. Write a PL/SQL Code Bind and Substitution Variables. Printing in PL/SQL
3. Write a PL/SQL block using SQL and Control Structures in PL/SQL
4. Write a PL/SQL Code using Cursors, Exceptions and Composite Data Types
5. Write a PL/SQL Code using Procedures, Functions, Triggers and Packages

TEXT BOOKS:

1 Fundamentals of Database Systems, Elmasri Navrate, 6th edition, Pearson Education

REFERENCE BOOKS:

1. "Database System Concepts", . Korth, Slberchatz, Sudarshan, 6th Edition, McGraw – Hill
2. Peter Rob and Carlos Coronel, " Database Systems Design, Implementation and Management", Thomson Learning, 5th Edition.
3. Introduction to Database Systems, CJ Date, Pearson
4. Data base Management Systems, Raghurama Krishnan, Johannes Gehrke, TATA McGraw Hill 3rd Edition

18A2105494- PYTHON PROGRAMMING LAB

Lecture – Tutorial-Practical:	0-1-2	Internal Marks:	40
Credits:	2	External Marks:	60

Prerequisites:

C Programming, Mathematics

Course Objectives:

- **To learn the fundamentals of python programming**
- **To get a solid understanding of python functions and data structures**
- To demonstrate the use of python lists and dictionaries.
- To implement methods and functions to improve readability of programs.
- Students able to describe and apply object-oriented programming methodology.
- Students able to build software for real needs and prior introduction to testing software.

Course Outcomes:

Upon successful completion of the course, the student will be able to:

CO1	Students able to experience with an interpreted Language and to build software for real needs.
CO2	Students able to use basic Decision structures, Boolean logic, variable types, assignments and operators.
CO3	Students able to describe and use of Python lists, dictionaries, tuples and sets.
CO4	To implement methods and functions to improve readability of programs.
CO5	Students able to describe and apply object-oriented programming methodology,top-down concepts in algorithm design.
CO6	Students should be able to design, code ,test and debug python language programs.

Contribution of Course Outcomes towards achievement of Program Outcomes (1 – Low, 2- Medium, 3 – High)

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO1	3	-	-	-	3	-	-	-	3	-	3	-
CO2	-	3	2	-	-	-	-	-	-	-	-	-
CO3	-	-	-	2	3	-	-	-	3	-	-	-
CO4	-	-	-	3	3	-	-	-	3	-	-	-
CO5	-	-	-	3	-	-		-	-	-	3	-
CO6	-	-	-	-	2	-	-	-	3	-	3	-

List of Experiments

Exercise 1 - Basics
a) Running instructions in Interactive interpreter and a Python Script
b) Write a program to purposefully raise Indentation Error and Correct it
Exercise 2 - Operations
a) Write a program to compute distance between two points taking input from the user
(Pythagorean Theorem)
b) Write a program add.py that takes 2 numbers as command line arguments and

prints its sum.
Exercise - 3 Control Flow
a) Write a Program for checking whether the given number is a even number or not.
b) Using a for loop, write a program that prints out the decimal equivalents of 1/2, 1/3, 1/4, . . . ,
1/10
c) Write a program using a for loop that loops over a sequence. What is sequence ?
d) Write a program using a while loop that asks the user for a number, and prints a countdown
from that number to zero.
Exercise 4 - Control Flow - Continued
a) Find the sum of all the primes below two million.
Each new term in the Fibonacci sequence is generated by adding the previous two terms. By
starting with 1 and 2, the first 10 terms will be:
1, 2, 3, 5, 8, 13, 21, 34, 55, 89, ...
b) By considering the terms in the Fibonacci sequence whose values do not exceed four million,
find the sum of the even-valued terms.
Exercise - 5 - DS
a) Write a program to count the numbers of characters in the string and store them in a
dictionary data structure
b) Write a program to use split and join methods in the string and trace a birthday with a
dictionary data structure.
Exercise - 6 DS - Continued
a) Write a program combine_lists that combines these lists into a dictionary.
b) Write a program to count frequency of characters in a given file. Can you use character
frequency to tell whether the given file is a Python program file, C program file or a text file?
Exercise - 7 Files
a) Write a program to print each line of a file in reverse order.
b) Write a program to compute the number of characters, words and lines in a file.
Exercise - 8 Functions
a) Write a function ball_collide that takes two balls as parameters and computes if they are
colliding. Your function should return a Boolean representing whether or not the balls are
colliding.
Hint: Represent a ball on a plane as a tuple of (x, y, r), r being the radius
If (distance between two balls centers) <= (sum of their radii) then (they are colliding)
b) Find mean, median, mode for the given set of numbers in a list.
Exercise - 9 Functions - Continued
a) Write a function nearly_equal to test whether two strings are nearly equal. Two strings a and b
are nearly equal when a can be generated by a single mutation on b.

b) Write a function dups to find all duplicates in the list.

c) Write a function unique to find all the unique elements of a list.

Exercise - 10 - Functions - Problem Solving

a) Write a function cumulative_product to compute cumulative product of a list of numbers.

b) Write a function reverse to reverse a list. Without using the reverse function.

c) Write function to compute gcd, lcm of two numbers. Each function shouldn't exceed one line.

Exercise 11 - Multi-D Lists

a) Write a program that defines a matrix and prints

b) Write a program to perform addition of two square matrices

c) Write a program to perform multiplication of two square matrices

Exercise - 12 - Modules

a) Install packages requests, flask and explore them. using (pip)

b) Write a script that imports requests and fetch content from the page. Eg. (Wiki)

c) Write a simple script that serves a simple HTTPResponse and a simple HTML Page

Exercise - 13 OOP

a) Class variables and instance variable and illustration of the self variable

i) Robot

ii) ATM Machine

Exercise - 14 GUI, Graphics

1. Write a GUI for an Expression Calculator using tk

2. Write a program to implement the following figures using turtle

Exercise - 15 - Testing

a) Write a test-case to check the function even_numbers which return True on passing a list of all

even numbers

b) Write a test-case to check the function reverse_string which returns the reversed string

Exercise - 16 - Advanced

a) Build any one classical data structure.

b) Write a program to solve knapsack problem.

TEXT BOOKS:

1 A Modern Approach, Vamsi Kurama, Pearson
2.Learning Python, Mark Lutz, Orielly

REFERENCE BOOKS:

Think Python, Allen Downey, Green Tea Press
Core Python Programming, W.Chun, Pearson.
Introduction to Python, Kenneth A. Lambert, Cengage

18A2100601- Professional Communication Skills

Lecture – Tutorial-Practical:	2-0-0	Internal Marks:	40
Credits:	1	External Marks:	60

Prerequisites:

Basic understanding of English

Course Objectives:

- This course is expected to relate Soft Skills to real life scenario besides real time experiences. This provides an additional support in training students on already available   practices. Attempt can be made to combine theory and practice enabling students learn more from experience.

- This course is expected to bridge the gap between theoretical knowledge and real time experiences while encountering challenges in the society.

- It is believed that the topics included in the course would enable the learner to develop a new perspective, to overcome a few current problems at the individual level and societal level.

Course Outcomes:

Upon successful completion of the course, the student will be able to:

CO1	Develop effective familiarity with soft skills along with right attitude to eliminate conflict and strife in their presentation skills.
CO2	Develop awareness on setting suitable goals and planning accordingly by using the techniques of time management to climb the ladder of success.
CO3	Enhance the learners' analytical and logical skills besides lateral thinking and kindle the true professional spirit.
CO4	Interpret any situation positively by managing stress at all scenarios.
CO5	Become a responsible citizen by imbibing social etiquette and ethics.
CO6	Draft appropriate written documents using the acquired knowledge on writing.

UNIT I: KNOW YOUR SOFT SKILLS

Know your Soft Skills: What are soft skills?, Soft Skills vs Hard Skills, Why Soft Skills are important for success?, Can Soft Skills be cultivated?

Writing Skills: Creative Writing, Summarization of Mini Projects or Paper Presentations

UNIT II: ATTITUDE AND EMOTIONAL INTELLIGENCE

Attitude: What is attitude? Benefits of Positive Attitude and How to build Positive Attitude?

Emotional Intelligence: What is Emotional Intelligence? How to balance Emotional Intelligence? and Crisis Management.

Writing Skills: Memo Writing, Writing Technical Abstracts

UNIT III: GOAL SETTING AND ADAPTABILITY
Goal Setting: What is a Goal? What are smart goals? Goal as Commitment. Steps to reach one's goals. **Adaptability:** What is Adaptability? What is its necessity? Adaptability as a tool to capture opportunities. **Writing Skills**: Writing Circulars, Writing Agenda and Minutes of Meeting
UNIT IV: TIME MANAGEMENT AND SOCIAL CONSCIOUSNESS
Time Management: What is Time Management? Time Stealers. Strategies for effective Time Management. Time Management and Goal Setting go hand in hand. **Social Consciousness:** Social Awareness and Civic Responsibility, Case Study based on Social Intelligence. **Writing Skills**: Technical Reports, Writing Reviews of Projects or any Technical Topic
TEXT BOOKS:
1. Pillai Sabina, Fernandez Agna, Soft Skills and Employability Skills, Cambridge Publishers. 2. Khera Shiv, You Can Win, Bloomsbury India, 1998. 3. Covey Sean, Seven Habits of Highly Effective Teens, New York, Fireside Publiishers, 1998. 4. Thomas A Harris, I am OK, You Are OK, New York-Harper And Row, 1972 5. Daniel Coleman, Emotional Intelligence, Bantam Book, 2006. 6. Carnegie Dale, How to Win Friends and Influence People, New York; Simon & Schuster, 1998.

18A2100602- Visual Communications

Lecture – Tutorial-Practical:	2-0-0	Internal Marks:	40
Credits:	1	External Marks:	60
Prerequisites:			
Basic understanding about communication components			
Course Objectives:			

The core objective of the paper is to create awareness about the theories and technical aspects of visualization process. The paper deals with visual communication theories and analysis of visuals. It also imparts knowledge of making visuals, video production skills and application of the software required for the purpose.

Unit – I of the syllabus focuses on understanding visuals based on theories and principles.

Unit – II introduces the techniques of shooting, framing, composition of visuals with still-camera and other techniques of visualization.

Unit – III imparts knowledge about planning and production techniques of camera, editing and post production.

Unit – IV introduces the basic concepts of software and its application required for making visual presentation.

Course Outcomes:

Upon successful completion of the course, the student will be able to:

CO1	The students will acquire the fundamental and foundation knowledge on image and creating meaning through visuals. They will acquire skills to present ideas and design messages.
CO2	They will be familiarised with communication and presentation of ideas visually.
CO3	The students will study various theories and concepts in understanding the visuals. They will learn new ways of understanding and presentation for different purposes.
CO4	The students will learn the key aspects of visual aspects in art, camera, and filming as well as trade secrets in communication media business to gain an up-to-date understanding of the field.
CO5	After studying the course, the students will acquire production planning skills and application of technology for projects based on communication.

Contribution of Course Outcomes towards achievement of Program Outcomes (1 – Low, 2- Medium, 3 – High)

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO1	3	-	2	-	3	3	2	2	1	1	1	1
CO2	3	-	2	-	3	3	2	2	1	1	1	1
CO3		-	2	-	-	-	-	-	-	1	1	1
CO4	3	-	1	3	3	-	-	1	1	1	1	1
CO5	2	-	1	-	3	3	2	2	1	1	1	1

CO6	3	-	2	-	3	3	2	2	1	1	1	1

UNIT- 1

Introduction to Visual Communication – Image and understanding – Semiotics – Signs, symbols, codes and meanings – Description of signs – Denotations and connotations – Theories of Visual communication – Elements of Visual Communication – Colour, form, depth, movement.

UNIT- 2

Still and Video images – Still Camera – Types of cameras – Different aspects of camera – Basic shots – Angles – Photo framing and composition – Camera movements – Lenses – Filters – Lighting – Technical and ethical perspectives of camera

UNIT- 3

Television camera Shots & angles – movements – mountings of video camera, Shooting – Lighting – Planning – Pre-production – Concept / story development – Script / Screenplay writing – Casting – Budget and Finance – Direction & Cinematography – Post-Production – Sound recording – Dubbing – Special effects and graphics – Final mixing

UNI 4

Audio Sound Editing software: Sound Forge **Video editing Software**: Adobe Premiere

TEXT BOOKS:

References:

1. Visual Communication – Images with messages 3rd Edition, Paul MartinLester, Thomson
    Wadsworth, USA 2003.
2. Herbert Zettl, Television Production Handbook, California, Thompson Wadsworth

Evaluation

At the end of the course students shall produce and submit a video capsule by applying the techniques which they learn based on the theoretical concepts.

Equipment required

Still cameras
1. Nikon D5600 Digital camera 18-55mm around Rs 40,000
2. Nikon D5600 with AF-P 70-300mm around Rs 50,000

Tripods

Digitek Professional Tripod around 6,000 (2min)

Video cameras
1. Canon XA11 Professional Camcorder around 80,000
2. Sony HXR-MC 2500 Camcorder around 75,000
3. Panasonic HC-MDH3GW professional Camcorder around Rs 65-75,000

18A2100603- Sanskrit

Lecture – Tutorial-Practical:	2-0-0	Internal Marks:	40
Credits:	1	External Marks:	60
Prerequisites:			
Basic understanding of languages			
Course Objectives:			

1.Reading, Writing, understanding and conversational skills of Sanskrit language will be imparted (through lecture method, Questions and Answers, Test, Open text book study, Role play, Discussion, Debate or collaboration assignment or case study).
2. (As Sanskrit enriched all most all Indian Languages through it's rich vocabulary) proper understanding of one's mother tongue and usage are expected.
3. To enable the students for the proper understanding of the culture, Heritage, Traditions, Thinking, ethics and values of our country Bharath.

Course Outcomes:

Upon successful completion of the course, the student will be able to:

CO1	Reading, Writing, understanding and conversational skills are developed.
CO2	Unity in diversity of our country is well understood.
CO3	The origin and development of Indian Languages is well understood
CO4	Proper usage of Language is achieved.
CO5	Sanskrit words that are familiar to us which we are using in our languages are identified.
CO6	The great Indian culture roots are well observed.

UNIT	TOPIC NAME	POEMS
I	विदुरानीतिः (Vidhuraneethi)	1 - 16
II	भर्तृहरि नीतिशतकम् (Bhartruhari – Neetisatakam)	1-16
III	संभाषणसंस्कृतमु (Prescribed Text by NRIIT)	1-16 Lesson From
IV	संस्कृतभाषा कौरालमु (Prescribed Text by NRIIT)	17-32 Lessons From संस्कृतमनोरम

Reference Books:

1. विदुरनीतिः from श्रीमन्महाभारतम् of वेदव्यासः
2. नीतिशतकम् of भर्तृहरिः
3. "TEACH YOURSELF SANSKRIT" (Graded Text Books) published by Rashtriya Sanskrit Sansthan, MHRD, Govt. of India, New Delhi.

18A2100604- Psychology

Lecture – Tutorial-Practical:	2-0-0	Internal Marks:	40
Credits:	1	External Marks:	60
Prerequisites:			
Basic understanding about communication components			

Course Objectives:

1. Aware of different applications of psychology to everyday life.
2. Aware of different work place issues, behavioral issues
3. Understand how the knowledge gained from this course can be used in their personal and professional lives.

Course Outcomes:

Upon successful completion of the course, the student will be able to:

CO1	The student will be able to understand the inter relationship of knowledge and our fund of knowledge
CO2	The students develops the discrimination between true and false knowledge
CO3	The students develops moral sense of Indian society
CO4	The students extends his mental horizons in understanding different stands of moral order
CO5	The students intuitively grasps the ways of understanding the world and our environment
CO6	The students gains an insight into the very nature of Science and Technology

Contribution of Course Outcomes towards achievement of Program Outcomes (1 – Low, 2- Medium, 3 – High)

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO1	1	2	-	-	-	2	-	3	-	-	-	2
CO2	1	2	-	2	-	2	-	2	-	-	-	2
CO3	1	-	-	-	-	3	-	3	-	-	-	-
CO4	1	-	-	-	-	3	-	3	-	-	-	-
CO5	1	-	=	-	-	2	-	-	-	-	-	2
CO6	3	3	3	3	3	3	3	-	-	3	3	3

UNIT- 1: Introduction

Psychology as a study of human behavior

Scope and fields of psychology

- Goal setting
- Time management

UNIT- 2 - Communication skills
- Non verbal communication
- Interpersonal skills
- Intrapersonal skills

UNIT- 3 - Life skills
- Emotional Intelligence
- Building resilience
- Stress management
- Mind management

UNIT 4 - Career skills
- Employability skills
- Presentation skills
- Leadership skills
- Team building
- Career planning

TEXT BOOKS:

1. Introduction to Psychology – N.L. Munn
2. Emotional Intelligence – Daniel Goleman

18A2100605- Engineering Philosophy

Lecture – Tutorial-Practical:	2-0-0	Internal Marks:	40
Credits:	1	External Marks:	60

Prerequisites:

Basic understanding

Course Objectives:

- The student gains knowledge and understanding of foundations of system building especially Indian philosophical systems i.e., darsanas, which originate from profound and novel ideas.
- It enables the student to understand the role of rigorously scrutinizing the internal structure, rigor, nature and validity of argumentation and thesis.
- It helps the student get training in the generation of new ideas, formulation of thoughts and its application for the benefit of society.
- The student is able to relate philosophy both Indian and western to literature, culture, society and lived experience.

Course Outcomes:

Upon successful completion of the course, the student will be able to:

CO1	The course in **Philosophy** is expected to bridge the gap between theory and practice by making the courses interactive.
CO2	The course of philosophy, which includes ethics and values, different ways of knowing, self and society, theory and practice, self and transcendence. These would enable the student in gaining knowledge about one's meaning and purpose of life, identifying one's goals and purpose and working out one's program of action in life
CO3	The course can provide a new understanding, based on which one can move to overcome the current problems, both at the individual level as well as at the societal level.

Unit I- Characteristic Features of Philosophy

(a) Structure and the content of the Vedas
(b) The essence of the fundamental Upanishads about the atman and Brahman
(c) A brief survey of heterodox and orthodox systems of Indian philosophy
(d) Greek Philosophy--Thales, Heraclitus, Parmenides, Democritus and Leucippus, Socrates, Plato and Aristotle

Unit II –Knowledge and Its Sources

(a)The difference between knowledge (*Vidyā*) and Ignorance (*Avidyā*)
(b)Sources of knowledge in Indian philosophy
(c) Language: Word as the root of knowledge (Bhartrahari's *Vākyapadīyam*)
(d)Socratic Method of knowledge as recollection
(e)Plato's metaphor of the divided line

Unit III Standards of Morality

(a) The subject-matter of ethics
(b) *Puruṣārthas* (*dharma, artha, kāma, and mikṣa)*
(c) Law of *karma*
(d) Western Ethics: Hedonism, Utilitarianism, Categorical Imperative

Unit IV Knowledge as Invention

(a) Caraka, Suṣruta Aryabhatta, Bhasakra
(b) Galileo, Copernicus, Kepler, Newton, Einstein

TEXT BOOKS:

C.D. Sharma, *A Critical Survey of Indian Philosophy*, Delhi: Motilal Banarsidass, 2000.

P.T.Raju, *Structural Depths of Indian Thought,* Delhi: South Asia Publishers,1985.

Frank Thilly, *History of Philosophy,* USA: Sagwan Press, 2015.

W.T.Grace*, A Critical History of Greek Philosophy,* Oregon: The Foating Press, 2010.

Bal Ram Singh (ed.), *Science and Technology in Ancient Indian Texts,* New Delhi: D.K. Printworld, 2012.

18A2100802- PROFESSIONAL ETHICS AND HUMAN VALUES

Lecture – Tutorial-Practical:	0-2-0	**Internal Marks:**	40
Credits:	0	**External Marks:**	60*

Prerequisites:
Basic understanding about Engineering profession

Course Objectives:
• To create awareness on engineering ethics and human values.   • To understand social responsibility of an engineer.   To instill moral and social values and loyalty.

Course Outcomes:
Upon successful completion of the course, the student will be able to:

CO1	Grooms themselves as ethical, responsible and societal beings.
CO2	Discuss ethics in society and apply the ethical issues related to engineering.
CO3	Exhibit the understanding of ethical theories in professional environment.
CO4	Recognize their role as social experimenters (engineers) and comprehend codes of ethics.
CO5	Identify the risks likely to come across in the professional world, analyzing them and find solutions.
CO6	Realize the responsibilities and rights of engineers in the society.

Contribution of Course Outcomes towards achievement of Program Outcomes (1 – Low, 2- Medium, 3 – High)

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO1	-	-	-	-	-	1	1	2	-	-	-	1
CO2	-	-	-	-	-	1	1	2	-	-	-	1
CO3	-	-	-	-	-	1	1	2	-	-	-	1
CO4	-	-	-	-	-	1	1	2	-	-	-	1
CO5	-	-	-	-	-	1	1	2	-	-	-	1
CO6	-	-	-	-	-	1	1	2	-	-	-	1

UNIT I
Human Values: Objectives, Morals, Values, Ethics, Integrity, Work ethics, Service learning , Virtues, Respect for others, Living peacefully, Caring, Sharing, Honesty, Courage, Valuing time,Cooperation, Commitment, Empathy, Self-confidence, Challenges in the work place.

UNIT II
Engineering ethics   Senses of 'Engineering Ethics' – Variety of moral issues – Types of inquiry – Moral dilemmas – Moral Autonomy – Kohlberg's theory – Gilligan's theory – Consensus and Controversy – Models of professional roles – Theories about right action – Self-interest – Customs and Religion – Uses of Ethical Theories.

UNIT III

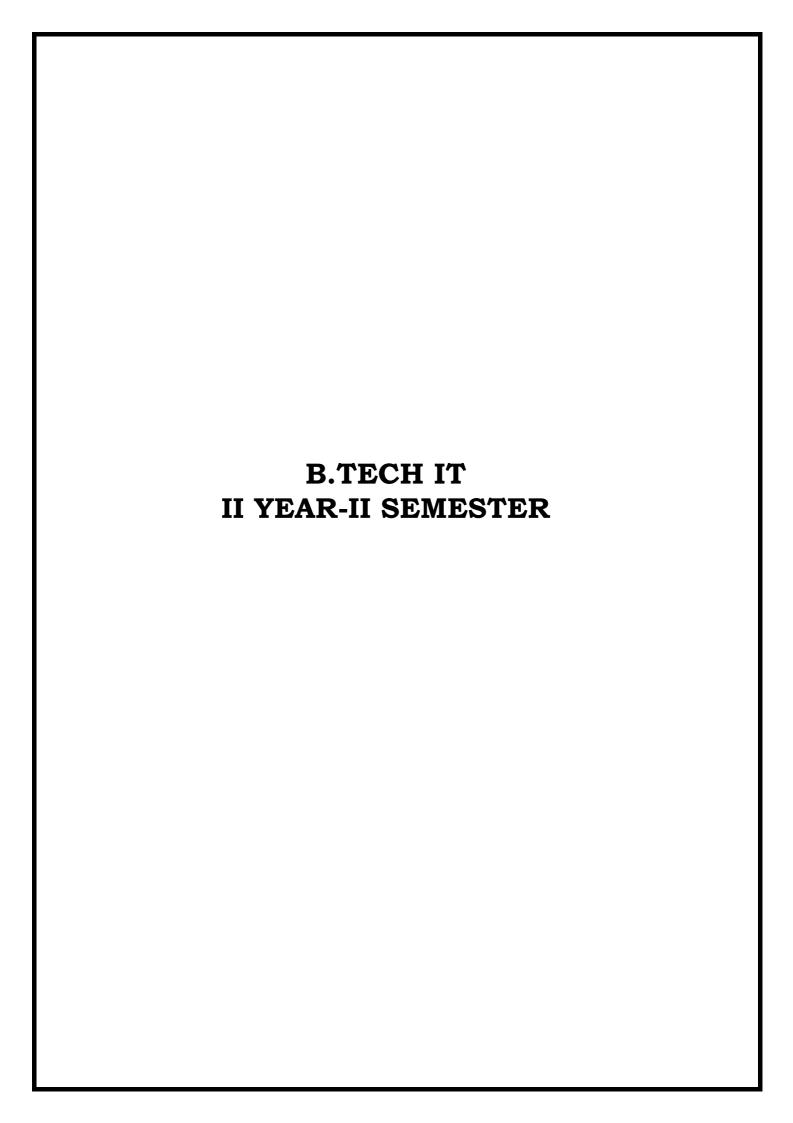| |
|---|
| **Engineering as Social Experimentation**: Engineering as experimentation, Engineers as responsible experimenters, Codes of ethics, Industrial standards, A balanced outlook on law, Case study: The challenger. |
| **UNIT IV** |
| **Safety, Responsibilities and Rights**: Safety and risk, types of risks, Assessment of safety and risk, Safe exit, Risk-benefit analysis, safety lessons from 'the challenger' , Case study: Power plants, Collegiality and loyalty, Collective bargaining, Confidentiality, Conflict of interests, Occupational crime, whistle blowing, Intellectual property rights, professional rights. |
| **TEXT BOOKS:** |
| A Text book on Professional Ethics and Human Values by R.S Naagarazan- New Age International Publishers. " Engineering Ethics includes Human Values" by M. Govindarajan, S. Natarajan and V. S. Senthil Kumar- PHI Learning Pvt. Ltd-2009 |
| **REFERENCE BOOKS:** |
| "Professional Ethics and Human Values" by A. Alavudeen, R. Kalil Rahman and M. Jayakumaran- Laxmi Publications. |
| **E-RESOURCES:** |
| www.onlineethics.org www.nspe.org www.globalethics.org www.ethics.org |

B.TECH IT
II YEAR-II SEMESTER

18A220020-PROBABILITY AND STATISTICS

Lecture – Tutorial:	3-1	Internal Marks:	40
Credits:	4	External Marks:	60

Prerequisites:

Course Objectives:

1.To familiarize the techniques in central tendency, curve fitting ,correlation and regression.
2. To familiarize the techniques in probability and random variables.
3. To familiarize the techniques in probability distribution.
4. To familiarize the techniques in large and small sample tests.
5.To equip the students to solve problems in their disciplines.

Course Outcomes:

Upon successful completion of the course, the student will be able to:

CO1	Student will be able to **Find the measures of central tendency and relation between them.(L1)**
CO2	Student will be able to **Evaluate the correlation coefficient, rank coefficient and regression.(L5)**
CO3	Students will be able to **Understand probabilities of events and expectationsof random variables for elementary problems.(L2)**
CO4	Students will be able to **Solve problems related to binomial and passion distribution.(L3)**
CO5	Student will be able to Compare situations in which it is appropriate to consider the relevance of the Normal distribution.(L4)
CO6	Student will be able to Construct hypothesis and carryout appropriate tests to checks its acceptability.(L3)

Contribution of Course Outcomes towards achievement of Program Outcomes (1 – Low, 2- Medium, 3 – High)

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO1	3	3	2	2	--	--	--	--	--	--	--	--
CO2	3	3	2	2	--	--	--	--	--	--	--	--
CO3	3	3	2	2	--	--	--	--	--	--	-	--
CO4	3	3	2	2	--	--	--	--	--	--	-	--
CO5	3	3	2	2	--	--	--	--	--	--	--	--
CO6	3	3	2	2	--	--	--	--	--	--	--	--

UNIT I

Unit 1: Descriptive statistics and methods for data science

(Pre-requisite:Data science, Statistics Introduction, Population vs Sample, Collection of data, primary and secondary data, Type of variable: dependent and independent Categorical and Continuous variables, Data visualization.---No Question selects from the above part)

Measures of Central tendency: Arithmetic Mean – Median – Mode - Geometric Mean-Harmonic Mean and Relations between them- Merits and Demerits.

Measures of Dispersion: Range – Quartile Deviation – Variance, Standard Deviation – Skewness - Kurtosis.

Curve Fitting and Principles of Least Squares.

Correlation- correlation coefficient - rank correlation - Regression coefficients Regression lines.

UNIT 2: Probability

Probability, probability axioms, addition law and multiplicative law of probability, conditional probability, Baye's theorem, random variables (discrete and continuous), probability density functions, properties, mathematical expectation.

Unit3: Distributions

Probability distribution - Binomial, Poisson approximation to the binomial distribution and normal distribution-their properties.

Unit 4: Estimation and Testing of hypothesis:Large sample tests Small sample tests

Population, sample distribution of mean, point estimation of mean and variance, confidence limits and intervals for mean, standard error, sample distribution of variance.

Formulation of null hypothesis, alternative hypothesis, the critical and acceptance regions, level of significance, two types of errors and power of the test.

Large Sample Tests: Test for single proportion, difference of proportions, test for single mean and difference of means. Confidence interval for parameters in one sample and two sample problems

Small Sample Tests: Student t-distribution (test for single mean, two means and paired t-test), testing of equality of variances (F-test), $\chi 2$ - test for goodness of fit, $\chi 2$ - test for independence of attributes.

Textbooks:

1. Miller and Freund, Probability and Statistics for Engineers,7/e, Pearson, 2008.
2. S.C. Gupta and V.K. Kapoor, Fundamentals of Mathematical Statistics, 11/e, Sultan Chand & Sons Publications, 2012.

Reference Books:

1. S. Ross, a First Course in Probability, Pearson Education India, 2002.
2. W. Feller, an Introduction to Probability Theory and its Applications, 1/e, Wiley, 1968.

18A2205401- WEB TECHNOLOGIES AND ADVANCED JAVA PROGRAMMING

Lecture – Tutorial-Practical:	3-0-0		Internal Marks:	40
Credits:	3		External Marks:	60*
Prerequisites:				

Course Objectives:

To understand the concepts of HyperText Markup Language and Cascading Style Sheets
To learn JavaScript for creating dynamic websites
To learn JavaScript for creating dynamic websites
To learn Server-Side Programming using Servlets and Java Server Pages.
To learn the creation of pure Dynamic Web Application using JDBC

Course Outcomes:

Upon successful completion of the course, the student will be able to:

CO1	Student able to Implement and design web based applications using features of HTML
CO2	Implement web based applications using features of XML
CO3	Student will Apply the concepts of server side technologies for dynamic web applications
CO4	Ability to design the web based applications using effective data base access with rich client interaction
CO5	Ability to Develop reusable component for Graphical User Interface applications

Contribution of Course Outcomes towards achievement of Program Outcomes (1 – Low, 2- Medium, 3 – High)

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO1	-	-	3	-	-	-	-	-	-	-	-	2
CO2	-	-	3	-	-	-	-	-	-	-	-	2
CO3	3	-	2	-	-	-	-	-	-	-	-	2
CO4	-	3	3	2	-	-	-	-	-	-	-	2
CO5	-	3	-	3	2	-	-	-	-	-	-	3

UNIT I

HTML tags, Lists, Tables, Images, forms, Frames, Cascading style sheets, Introduction to Java script, objects in Java Script, Dynamic HTML with Java Script.

UNIT II

Working with XML: Document Type Definition, XML schemas, Document object model, XSLT, DOM and SAX.

UNIT III

Web Servers and Servlets: Tomcat web server, Introduction to Servlets: Lifecycle of a Servlet, JSDK, The Servlet API, The javax.servlet Package, Reading Servlet parameters, and Reading Initialization parameters. The javax.servlet HTTP package, Handling Http Request & Responses, Using Cookies-Session Tracking, Security Issues.

UNIT IV

Database Access: Database Programming using JDBC, studying javax.sql.* package, accessing a database from a JSP page,

Introduction to JSP: The Problem with Servlet. The Anatomy of a JSP Page, JSP Processing. JSP application design with MVC

JSP Application Development: Generating Dynamic Content, Using Scripting ElementsImplicit JSP Objects, Conditional Processing – Displaying Values Using an Expression to Set an Attribute, Declaring Variables and Methods Error Handling and Debugging Sharing Data Between JSP pages, Requests, and Users Passing Control and Date between Pages – Sharing Session and Application Data – Memory Usage Considerations

TEXT BOOKS:

- The Complete Reference, Java 2 , 3ed, Patrik Naughton, Herbert Schildt, TMH
- Programming the World Wide Web, Robet W Sebesta, 7ed, Pearson.
- Web Technologies, Uttam K Roy, Oxford Java Server Pages , Hans Bergstan, Oreilly

REFERENCE BOOKS:

- Web Technologies, HTML< JavaScript, PHP, Java, JSP, XML and AJAX, Black book, Dream Tech.
- An Introduction to Web Design, Programming, Paul S Wang, Sanda S Katila, Cengage
- An introduction to Web Design and Programming, Wang Thomson
- Web application technologies concepts, Knuckles, John Wiley.
- Programming world wide web, Sebesta, Pearson
- Beginning Web Programming, Jon Duckett ,Wrox, Wiley Java server pages, Pekowsky, Pearson

E-RESOURCES:

1. https://www.w3schools.com/
2. https://www.tutorialspoint.com/perl/
3. https://www.railstutorial.org/book
4. https://www.cs.usfca.edu/~galles/visualization/Algorithms.html

18A2205402- SOFTWARE ENGINEERING

Lecture – Tutorial- Practical:	3-0-2*	Internal Marks:	40
Credits:	5	External Marks:	60*
Prerequisites:			

Course Objectives:

To understand the software requirements and SRS document

To design and develop correct and robust software products.

To understand the quality control and how to ensure good quality software.

To understand the planning and estimation of software projects.

To understand the implementation issues, validation and verification procedures

Course Outcomes:

Upon successful completion of the course, the student will be able to:

CO1	Understand the basic concepts of Software engineering, applications, agile development and compare different software process models.
CO2	Analyze the principles of requirement engineering
CO3	Create architectural design for a given project.
CO4	Apply different testing techniques

Contribution of Course Outcomes towards achievement of Program Outcomes (1 – Low, 2- Medium, 3 – High)

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO1	3	3	3	3	3	--	--	2	2	2	3	2
CO2	3	3	3	3	3	--	--	2	2	2	3	2
CO3	3	3	3	3	3	--	--	2	2	2	3	2
CO4	3	3	3	3	3	--	--	2	2	2	3	2

UNIT I

Software and Software Engineering: The Nature of Software, Defining Software, Software Application Domains, Legacy Software, The Unique Nature of Web Apps, Software Engineering, The Software Process, Software Engineering Practice, The Essence of Practice, General Principles, Software Myths.

The Software Process: Process Models, A Generic Process Model, Process Assessment and Improvement, Prescriptive Process Models, Specialized Process Models, The Unified Process, Personal and Team Process Models, Process Technology, Product and Process.

Agile Development: What Is Agility? Agility and the Cost of Change, What Is an Agile Process? Extreme Programming (XP). Other Agile Process Models, A Tool Set for the Agile Process.

Practice Session:

1. Write down the problem statement for a suggested system of relevance

UNIT II

Understanding Requirements: Requirements Engineering, Establishing the Groundwork, Eliciting Requirements, Developing Use Cases, Building the Requirements Model, Negotiating Requirements, Validating Requirements.

Requirements Modeling: Scenarios, Information and Analysis classes: Requirements Analysis, Scenario-Based Modeling, UML Models That Supplement the Use Case, Data Modeling Concepts, Class-Based Modeling.

Requirements Modeling: Flow, Behavior, Patterns, And Web apps: Requirements Modeling Strategies, Flow-Oriented Modeling, Creating a Behavioral Model, Patterns for Requirements Modeling, Requirement modeling for WebApps.

     1. Do requirement analysis and develop Software Requirement Specification Sheet (SRS) for suggested system.

UNIT III

Design Concepts: Design within the Context of Software Engineering, the Design Process, Design Concepts, the Design Model. **Architectural Design**: Software Architecture, Architectural Genres, Architectural Styles, Architectural Design, Assessing Alternative Architectural Designs.

Modeling Component-Level Design: What Is a Component? Designing Class-Based Components, Conducting Component Level Design, and Component level design for Web Apps.

Performing User Interface Design: The Golden Rules, User Interface Analysis and Design, Interface Analysis, Interface Design Steps.

Practice Session:

    1. Preparation of Design Documents for suggested system.

    2. Study and usage of any Design phase CASE tool.

UNIT IV

Software Testing Strategies: A Strategic Approach to Software Testing, Strategic Issues, Test Strategies for Conventional Software, Test Strategies for Object-Oriented Software, Validation testing, System testing, the art of debugging.

Testing Conventional Applications: Software Testing Fundamentals, Internal and External Views of Testing, White Box Testing, Basis Path Testing, Control Structure Testing, Black-Box Testing, Model-Based Testing, Testing for Specialized Environments, Architectures, and Applications, Patterns for Software Testing.

 Practice Session:

1. Preparation of Testing Phase related documents for some problems
2. To perform unit testing and integration testing
3. To perform various white box and black box testing techniques.

TEXT BOOKS:

Roger S.Pressman, "Software Engineering- A Practitioner's Approach". Tata McGraw-Hill International 7th ed, 2010

REFERENCE BOOKS:

Ian Somerville, "Software Engineering". 9th ed, Pearson Education. 2011.
Carlo Ghezzi, Mehdi Jazayeri and Dino Mandrioli, "Fundamentals of Software Engineering".2 ed, PHI. 2009
RajibMall, Fundamentals of Software Engineering. 3 ed, PHI. 2009.
Software Engineering, A Precise Approach, Pankaj Jalote, Wiley India,2010.

E-RESOURCES:

1. http://www.dcnicn.com/BusinessNews/WVU-MIS13Apr00/Software-Engineering.pdf
2. http://www.comp.lancs.ac.uk/computing/resources/IanS/SE7/Presentations/PDF/ch1.pdf
3. http://sites.computer.org/ccse/SE2004Volume.pdf
4. http://homepages.cs.ncl.ac.uk/brian.randell/NATO/nato1968.PDF
5. http://www.dau.mil/pubs/pdf/SEFGuide 01-01.pdf

18A2205403- COMPUTER ORGANIZATION

Lecture – Tutorial-Practical:	3-0-0	Internal Marks:	40
Credits:	3	External Marks:	60*
Prerequisites:			

Course Objectives:
Understand the architecture of a modern computer with its various processing units. Also the Performance measurement of the computer system.
In addition to this the memory management system of computer.

Course Outcomes:

Upon successful completion of the course, the student will be able to:

CO1	Able to understand the basic components and the design of CPU, ALU and Control unit
CO2	Students can calculate the effective address of an operand by addressing modes
CO3	Ability to understand memory hierarchy and its impact on computer cost/performance..
CO4	Ability to understand the advantage of instruction level parallelism and pipelining for high performance Processor design.

Contribution of Course Outcomes towards achievement of Program Outcomes (1 – Low, 2- Medium, 3 – High)

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO1	3	3	--	--	--	--	--	--	--	--	--	--
CO2	3	3	--	3	--	--	--	--	--	--	--	--
CO3	3	3	--	--	--	--	--	--	--	--	--	--
CO4	3	3	--	--	--	--	--	--	---	--	--	--

UNIT I

Basic Structure Of Computers : Computer Types, Functional unit, Basic Operational concepts, Bus structures, Software, Performance.

Register Transfer and Micro-Operations: Register Transfer Language, Register Transfer, Bus and memory Transfers, Arithmetic Micro-operations, Logic Microoperations, Shift Micro-operations, Arithmetic Logic Shift Unit.

Basic Computer Organization and Design: Instruction codes, Computer Registers,

Computer Instructions, Timing and Control, Instruction cycle, Memory Reference Instructions,

Input-Output and Interrupts

UNIT II

Central Processing Unit: General register Organization, Stack Organization, Instruction Formats, Addressing Modes, Data Transfer and Manipulation, Program Control, Reduced Instruction Set Computer (RISC).

Micro Programmed Control : Control memory, Address sequencing, microprogram example,

design of control unit.

UNIT III

Computer Arithmetic : Addition and subtraction, multiplication Algorithms, Division Algorithms, Floating – point Arithmetic operations.

Memory Organization: Memory Hierarchy, Main Memory, Auxiliary memory, Associative Memory, Cache Memory, Virtual Memory, Memory Management Hardware.**Input Output**

UNIT IV

Organization: Peripheral Devices, Input-output Interface, Asynchronous Data Transfer, Modes of Transfer, Priority Interrupt, Direct Memory Access (DMA), Input-Output Processor.,Serial Communication. Standard I/O Interfaces: PCI Bus, USB

Pipeline and vector processing: parallel processing, pipelining, Arithmetic pipeline, Instruction pipeline, RISC Pipeline, Vector Processing

TEXT BOOKS:

1 Morris M. Mano, Computer Systems Architecture.3 Ed, Pearson/PHI, 2013

[2] Carl Hamacher, Zvonko Vranesic, Safwat Zaky: Computer Organization, 5th Edition, Tata McGraw Hill, 2002.

REFERENCE BOOKS:

John P.Hayes, 'Computer architecture and Organisation', Tata McGraw-Hill,
Third edition, 1998.

E-RESOURCES:

https://www.tutorialspoint.com/computer_organization/index.asp

https://www.geeksforgeeks.org/computer-organization-basic-computer-instructions/

OE- MICROPROCESSOR AND ITS APPLICATIONS

Lecture – Tutorial-Practical:	3-0-0		Internal Marks:	40
Credits:	3		External Marks:	60*

Prerequisites:

Number systems & digital logic design concepts, Basic compilation process

Course Objectives:

To demonstrate various families of Intel microprocessors and Significance of 8086

To explain Assembly language program development tools.

To know the architecture details of 8051 microcontroller.

To study and design the concepts of Counters/timers and serial port of 8051.

To understand the architecture of ARM7 and its THUMB instruction set.

To discuss the interfacing of 8051 to various peripheral devices.

Course Outcomes:

Upon successful completion of the course, the student will be able to:

CO1	To Describe the basics of 8086 microprocessors architectures and its Functionalities
CO2	To Design and develop 8086 Microprocessor based systems for real time applications using low level language like ALP
CO3	To Analyze 8051 microcontrollers architectures and its functionalities
CO4	To Describe the importance of Timers/Counters and Serial ports of 8051 microcontroller
CO5	To Describe the basics of ARM and ARM7 architecture and its functionalities
CO6	To Interface external peripherals and I/O devices and program the 8051 microcontroller

Contribution of Course Outcomes towards achievement of Program Outcomes (1 – Low, 2- Medium, 3 – High)

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO1	3	3	-	-	-	-	-	-	-	-	-	-
CO2	3	3	2	-	3	-	-	-	-	-	2	-
CO3	3	2	3	2	2	-	-	-	-	-	2	-
CO4	2	2	2	-	-	-	-	-	-	-	-	-
CO5	3	2	2	2	3	-	-	-	-	-	2	-
CO6	2	2	2	3	2	-	-	-	-	-	-	-

UNIT I
UNIT-I

Microprocessor Architecture: Introduction to Microprocessors, Families of a Microprocessor, 8086 Microprocessor- Features, Architecture, Pin diagram of 8086.

8086 Architecture Modes: Register Organization, 8086 System Timing, Minimum Mode and Maximum Mode of Operation

UNIT II
UNIT-II **Instruction Set: A**ddressing Modes, Instruction Set, Assembler Directives, and Program development steps. **Programming and Interrupts:** Assembly Language Program Development Tools, Programs with an Assembler, Interrupt Structure, Interrupt Service Routine, Interrupt Vector Table,

UNIT III
UNIT-III **Microcontroller:** 8051 Microcontroller Architecture, Pin Diagram, Addressing Modes, Instruction Set and Programs, 8051 Memory and I/O Interfacing. **8051 Interfacing:** Modes of Timer Operation, Serial Port Operation, Interrupt Structure of 8051, Interfacing of Seven Segment Displays, Stepper Motor.

UNIT IV
UNIT-IV **ARM:** Introduction to 16/32 Bit Processor, Internal Architecture of ARM 7, Register Organization, ARM and THUMB Operating Modes, Development Tools. **Peripherals and Interfacing:** 8255 PPI – Various Modes of Operation and Interfacing to 8086, Keyboard and Seven Segment Displays, Stepper Motor, D/A and A/D Converter Interfacing.

TEXT BOOKS:
TEXT BOOKS: 1. Advanced Microprocessor and Peripherals (Architecture, Programming & Interfacing) by A.K. Ray & K.M. Bhurchandi – TMH Publication. 2. Microcontrollers [theory and applications] TMH publication by Ajay V. Deshmukh. 3. Microcontrollers: Architecture, Programming, Interfacing and System Design, 2nd Edition, by Raj Kamal, Pearson Publications.

REFERENCE BOOKS:
1. Microprocessor and Interfacing by Douglas Hall 2nd Edition. 2. The 8051 Microcontroller & Embedded Systems by Mazidi & Mazidi – Pearson / PHI publication.

E-RESOURCES:
https://freevideolectures.com/course/3018/microprocessors-and-microcontrollers/1
https://www.tutorialspoint.com/microprocessor
https://www.javatpoint.com/microprocessor

18A2205491- WEB TECHNOLOGIES AND ADVANCED JAVA PROGRAMMING LAB

Lecture – Tutorial- Practical:	0-0-3	Internal Marks:	40
Credits:	3	**External Marks:**	60*
Prerequisites:			

JAVA PROGRAMMING, DBMS

Course Objectives:

HyperText Markup Language (HTML) and Cascading Style Sheets (CSS) for laying out

(formatting) pages that contain text, images and graphics.

Extensible Markup Languages (XML is used to store and transport data among webpages), a

mechanism for defining new tag sets and interchanging data among web applications.

Client-side Programming using JavaScript for validating the data.

Server-Side Programming using servlets are to generate static content and Java

Server Pages are used to generate dynamic content.

Creating a pure Dynamic Web Application which retrieves the data from

Database according to the client request using JDBC.

Course Outcomes:
Upon successful completion of the course, the student will be able to:

CO1	Create a website statically or dynamically.
CO2	Get knowledge on displaying and decorating the contents in a webpage.
CO3	Learn the concepts of store and transport the data among webpages.
CO4	Generate static or dynamic content according to the client's request.
CO5	Create objects with which the client can communicate with server.
CO6	Provide User Authentication by using cookies and back end operations using JDBC.

Contribution of Course Outcomes towards achievement of Program Outcomes (1 – Low, 2- Medium, 3 – High)

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO1	3	3	--	--	--	--	--	--	--	--	--	--
CO2	3	3	--	3	--	--	--	--	--	--	--	3
CO3	3	3	--	--	--	--	--	--	--	--	--	--
CO4	3	3	--	--	--	--	--	--	---	--	--	3
CO5	3	3	--	--	--	--	--	--	--	--	--	3
CO6	3	3	--	--	--	--	--	--	--	--	--	--

Week-1:
Design the following static web pages required for an online book store web site.
1) **HOME PAGE:**

The static home page must contain three **frames**.

      Top frame :   Logo and the college name and links to Home page, Login page, Registration page,

                    Catalogue page and Cart page (the description of these pages will be given below).

      Left frame :   At least four links for navigation, which will display the catalogue of respective links.

                    For e.g.: When you click the link **"CSE"** the catalogue for **CSE** Books should be

                    displayed in the Right frame.

      Right frame: The *pages to the links in the left frame must be loaded here*. Initially this page contains

                    description of the web site.

Logo	Web Site Name			
Home	Login	Registration	Catalogue	Cart
CSE ECE EEE CIVIL	Description of the Web Site			

2) LOGIN PAGE:

This page looks like below:

Logo	Web Site Name			
Home	Login	Registration	Catalogue	Cart
CSE ECE EEE CIVIL	Login : [ ] Password: [ ] Submit Reset			

3) CATOLOGUE PAGE:

The catalogue page should contain the details of all the books available in the web site in a table.

The details should contain the following:

      1. Snap shot of Cover Page.

      2. Author Name.

      3. Publisher.

      4. Price.

      5. Add to cart button.

Note: Week 2 contains the remaining pages and their description.

Week-2:

4) CART PAGE: The cart page contains the details about the books which are added to the cart.

The cart page should look like this:

5) REGISTRATION PAGE:

Create a "*registration form* "with the following fields

      1) Name (Text field)

      2) Password (password field)

      3) E-mail id (text field)

      4) Phone number (text field)

      5) Sex (radio button)

      6) Date of birth (3 select boxes)

      7) Languages known (check boxes – English, Telugu, Hindi, Tamil)

      8) Address (text area)

WEEK 3:

VALIDATION:

Write *JavaScript* to validate the following fields of the above registration page.

      1.   Name (Name should contains alphabets and the length should not be less than 6 characters).

      2. Password (Password should not be less than 6 characters length).

      3. E-mail id (should not contain any invalid and must follow the standard pattern
            name@domain.com)

      4. Phone number (Phone number should contain 10 digits only).

      Note : You can also validate the login page with these parameters.

      Use PHP to connect with the database to store the above details.

Week-4:

Design a web page using **CSS (C**ascading **S**tyle **S**heets**)** which includes the following:

      1) Use different font, styles:

In the style definition you define how each selector should work (font, color etc.).
Then, in the body of your pages, you refer to these selectors to activate the styles.
For example:

```
<HTML>
<HEAD>
<style type="text/css">
B.headline {color:red; font-size:22px; font-family:arial; text-decoration:underline}
</style>

</HEAD>
```

```
<BODY>
<b>This is normal bold</b><br>
Selector {cursor:value}

For example:

<html>
<head>
<style type="text/css">
.xlink {cursor:crosshair}
.hlink{cursor:help}
</style>
</head>

<body>
<b>
<a href="mypage.htm" class="xlink">CROSS LINK</a>
<br>
<a href="mypage.htm" class="hlink">HELP LINK</a>
</b>
</body>
</html>

<b class="headline">This is headline style bold</b>
</BODY>

</HTML>
```

2) Set a background image for both the page and single elements on the page.
You can define the background image for the page like this:

```
BODY {background-image:url(myimage.gif);}
```

3) Control the repetition of the image with the background-repeat property.
As background-repeat: repeat
Tiles the image until the entire page is filled, just like an ordinary background image in plain HTML.

4) Define styles for links as
        A:link
        A:visited
        A:active
        A:hover
Example:
```
<style type="text/css">
A:link {text-decoration: none}
A:visited {text-decoration: none}
A:active {text-decoration: none}
A:hover {text-decoration: underline; color: red;}
</style>
```

5) Work with layers:
For example:
LAYER 1 ON TOP:

```
<div style="position:relative; font-size:50px; z-index:2;">LAYER 1</div>
<div style="position:relative; top:-50; left:5; color:red; font-size:80px; zindex:
1">LAYER 2</div>
LAYER 2 ON TOP:
<div style="position:relative; font-size:50px; z-index:3;">LAYER 1</div>
<div style="position:relative; top:-50; left:5; color:red; font-size:80px; zindex:
4">LAYER 2</div>
```

6) Add a customized cursor:

      Selector {cursor:value}

For example:

```
<html>
<head>
<style type="text/css">
.xlink {cursor:crosshair}
.hlink{cursor:help}
</style>
</head>

<body>
<b>
<a href="mypage.htm" class="xlink">CROSS LINK</a>
<br>
<a href="mypage.htm" class="hlink">HELP LINK</a>
</b>
</body>
</html>
```

Week-5:

Write an XML file which will display the Book information which includes the following:

      1) Title of the book
      2) Author Name
      3) ISBN number
      4) Publisher name
      5) Edition
      6) Price

Write a Document Type Definition (DTD) to validate the above XML file.

Display the XML file as follows.

The contents should be displayed in a table. The header of the table should be in color GREY. And the

Author names column should be displayed in one color and should be capitalized and in bold.

Use your own colors for remaining columns.

Use XML schemas XSL and CSS for the above purpose.

Note: Give at least for 4 books. It should be valid syntactically.

Hint: You can use some xml editors like XML-spy

Week-6:

1) Install TOMCAT web server and APACHE.

While installation assign port number 4040 to TOMCAT and 8080 to APACHE. Make sure that these ports are available i.e., no other process is using this port.

2) Access the above developed static web pages for books web site, using these servers by putting the web pages developed in week-1 and week-2 in the document root.

Access the pages by using the urls : http://localhost:4040/rama/books.html (for tomcat)

http://localhost:8080/books.html (for Apache)

Week-7:
User Authentication :

Assume four users user1,user2,user3 and user4 having the passwords pwd1,pwd2,pwd3 and pwd4 respectively. Write a servelet for doing the following.

1. Create a Cookie and add these four user id's and passwords to this Cookie.

2. Read the user id and passwords entered in the Login form (week1) and authenticate with the values (user id and passwords) available in the cookies.

If he is a valid user(i.e., user-name and password match) you should welcome him by name(user-name) else you should display " You are not an authenticated user ".

Use init-parameters to do this. Store the user-names and passwords in the webinf.xml and access them in the servlet by using the getInitParameters() method.

Week-8:

Install a database(Mysql or Oracle).

Create a table which should contain at least the following fields: name, password, email-id, phone number(these should hold the data from the registration form).

Practice 'JDBC' connectivity.

Write a java program/servlet/JSP to connect to that database and extract data from the tables and display them. Experiment with various SQL queries.

Insert the details of the users who register with the web site, whenever a new user clicks the submit button in the registration page (week2).

Week-9:

Write a JSP which does the following job:

Insert the details of the 3 or 4 users who register with the web site (week9) by using registration form. Authenticate the user when he submits the login form using the user name and password from the database (similar to week8 instead of cookies).

Week-10:

Create tables in the database which contain the details of items (books in our case like Book name , Price, Quantity, Amount)) of each category. Modify your catalogue page (week 2)in such a way that you should connect to the database and extract data from the tables and display them in the catalogue page using JDBC.

18A2205991--Aptitude and Reasoning

Lecture – Tutorial-Practical:	0-0-2	Internal Marks:	40
Credits:	1	External Marks:	60*

UNIT – I
1. PERCENTAGES
2. SIMPLE INTEREST & COMPOUND INTEREST
3. PROFIT AND LOSS

UNIT – II
1. TIME AND WORK
2. PIPES AND CISTERN
3. TIME, SPEED AND DISTANCE

UNIT- III
1. DATA INTERPRETATION
2. SYLLOGISMS, AVERAGES

UNIT- IV
1. VISUAL /DIAGRAMATIC REASONING
2. CODING AND DECODING
3. NUMBER SERIES
4. PROBLEMS ON AGES

TEXT BOOKS:
1) APTIPEDIA, WILEY
2) Quantitative Aptitude, RS AGARWAL, S.Chand Publishers

REFERENCE BOOKS:
1. HOW TO PREPARE FOR Quantitative Aptitude, ARUN SHARMA, Mc GRAW HILL

18A2205791--Mini project

Lecture – Tutorial-Practical:	0-0-2	Internal Marks:	40
Credits:	1	External Marks:	60*

The mini project is designed to help students develop practical ability and knowledge about practical tools/techniques in order to solve real life problems related to the industry, academic institutions and computer science research. The course Mini Project is one that involves practical work for understanding and solving problems in the field of computing. Any computer science project usually consists of the following: analysis, design, coding/implementation and testing of some information system or subsystem, such as, a piece of software. The subsystem does not have to be a computer program; a design document might be the appropriate output from a design study. The design and development of hardware system/subsystem would also be an appropriate project, however, in this course we expect a software system or subsystem. This course will also develop your investigative, research and report writing skills and will provide an opportunity for you, to investigate a chosen topic in considerable depth. Mini Project provides the opportunity for students to demonstrate the application of their programming and research skills, and to apply their knowledge to complex computing problems.
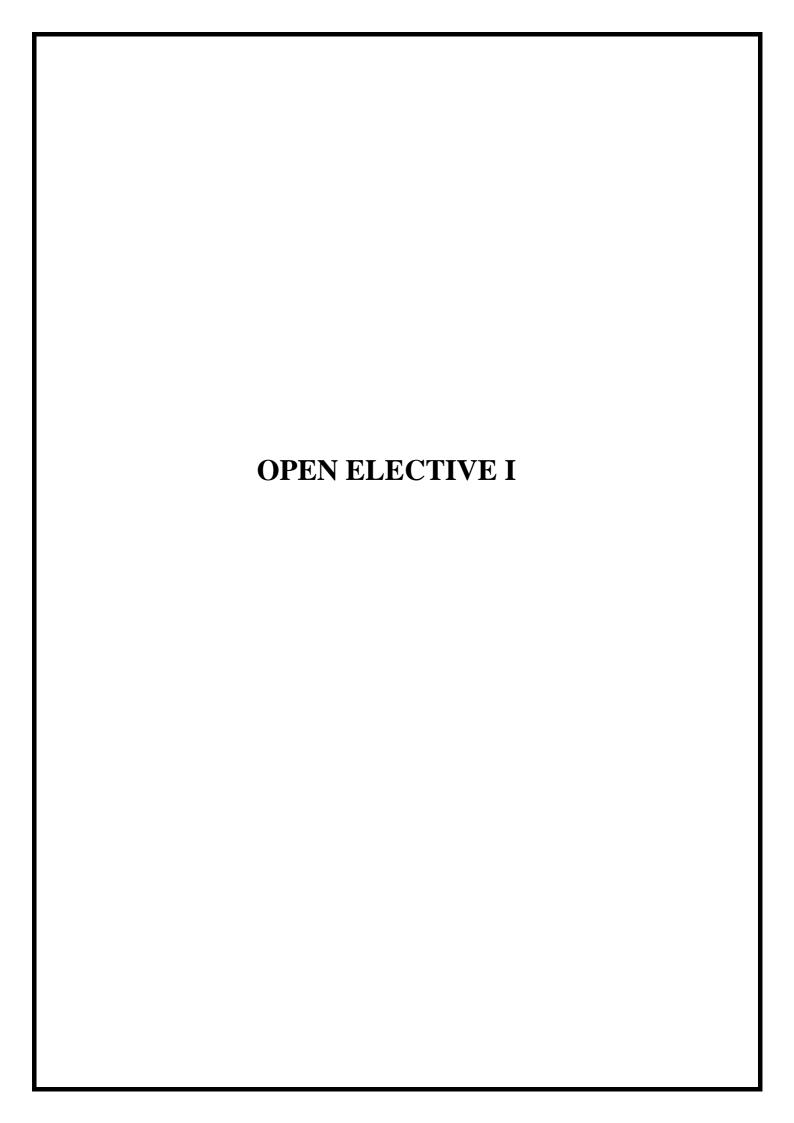
Students can take up small problems in the field of Computer Science and Engineering as mini project.

Project can be related to solution to an engineering problem, verification and analysis of experimental data available, conducting experiments on various engineering subjects, material characterization, studying a software tool for the solution of an engineering problem etc.

A batch of 3 students can form it as group
The type of the project selection could be an application, product, a review or a research work.
Project Review Committee will conduct 3 internal Reviews and one Final Review for 100 Marks.

Review I (Project Synopsis ,analysis)	20 Marks
Review II(Project Design)	20 Marks
Review III(Project Implementation)	20 Marks
Final Review (Project Evaluation with conclusion and report)	40 Marks

OPEN ELECTIVE I

18A2212601 - SOFTWARE ENGINEERING

Lecture – Tutorial- Practical:	3-0-2*	Internal Marks:	40
Credits:	5	External Marks:	60*
Prerequisites:			

Course Objectives:

To understand the software requirements and SRS document

To design and develop correct and robust software products.

To understand the quality control and how to ensure good quality software.

To understand the planning and estimation of software projects.

To understand the implementation issues, validation and verification procedures

Course Outcomes:

Upon successful completion of the course, the student will be able to:

CO1	Understand the basic concepts of Software engineering, applications, agile development and compare different software process models.
CO2	Analyze the principles of requirement engineering
CO3	Create architectural design for a given project.
CO4	Apply different testing techniques

Contribution of Course Outcomes towards achievement of Program Outcomes (1 – Low, 2- Medium, 3 – High)

	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO 10	PO 11	PO 12
CO1	3	3	3	3	3	--	--	2	2	2	3	2
CO2	3	3	3	3	3	--	--	2	2	2	3	2
CO3	3	3	3	3	3	--	--	2	2	2	3	2
CO4	3	3	3	3	3	--	--	2	2	2	3	2

UNIT I

Software and Software Engineering: The Nature of Software, Defining Software, Software Application Domains, Legacy Software, The Unique Nature of Web Apps, Software Engineering, The Software Process, Software Engineering Practice, The Essence of Practice, General Principles, Software Myths.

The Software Process: Process Models, A Generic Process Model, Process Assessment and Improvement, Prescriptive Process Models, Specialized Process Models, The Unified Process, Personal and Team Process Models, Process Technology, Product and Process.

Agile Development: What Is Agility? Agility and the Cost of Change, What Is an Agile Process? Extreme Programming (XP). Other Agile Process Models, A Tool Set for the Agile Process.

Practice Session:

1. Write down the problem statement for a suggested system of relevance

UNIT II

Understanding Requirements: Requirements Engineering, Establishing the Groundwork, Eliciting Requirements, Developing Use Cases, Building the Requirements Model, Negotiating Requirements, Validating Requirements.

Requirements Modeling: Scenarios, Information and Analysis classes: Requirements Analysis, Scenario-Based Modeling, UML Models That Supplement the Use Case, Data Modeling Concepts, Class-Based Modeling.

Requirements Modeling: Flow, Behavior, Patterns, And Web apps: Requirements Modeling Strategies, Flow-Oriented Modeling, Creating a Behavioral Model, Patterns for Requirements

Modeling, Requirement modeling for WebApps.

Practice Session:

1. Do requirement analysis and develop Software Requirement Specification Sheet (SRS) for suggested system.

UNIT III

Design Concepts: Design within the Context of Software Engineering, the Design Process, Design Concepts, the Design Model. **Architectural Design**: Software Architecture, Architectural Genres, Architectural Styles, Architectural Design, Assessing Alternative Architectural Designs.

Modeling Component-Level Design: What Is a Component? Designing Class-Based Components, Conducting Component Level Design, and Component level design for Web Apps.

Performing User Interface Design: The Golden Rules, User Interface Analysis and Design, Interface Analysis, Interface Design Steps.

Practice Session:

3. Preparation of Design Documents for suggested system.

4. Study and usage of any Design phase CASE tool.

UNIT IV

Software Testing Strategies: A Strategic Approach to Software Testing, Strategic Issues, Test Strategies for Conventional Software, Test Strategies for Object-Oriented Software, Validation testing, System testing, the art of debugging.

Testing Conventional Applications: Software Testing Fundamentals, Internal and External Views of Testing, White Box Testing, Basis Path Testing, Control Structure Testing, Black-Box Testing, Model-Based Testing, Testing for Specialized Environments, Architectures, and Applications, Patterns for Software Testing.

 Practice Session:
1. Preparation of Testing Phase related documents for some problems
2. To perform unit testing and integration testing
3. To perform various white box and black box testing techniques.

TEXT BOOKS:

Roger S.Pressman, "Software Engineering- A Practitioner's Approach". Tata McGraw-Hill International 7th ed, 2010

REFERENCE BOOKS:

Ian Somerville, "Software Engineering". 9th ed, Pearson Education. 2011.
Carlo Ghezzi, Mehdi Jazayeri and Dino Mandrioli, "Fundamentals of Software Engineering".2 ed, PHI. 2009
RajibMall, Fundamentals of Software Engineering. 3 ed, PHI. 2009.
Software Engineering, A Precise Approach, Pankaj Jalote, Wiley India,2010.

E-RESOURCES:

6. http://www.dcnicn.com/BusinessNews/WVU-MIS13Apr00/Software-Engineering.pdf
7. http://www.comp.lancs.ac.uk/computing/resources/IanS/SE7/Presentations/PDF/ch1.pdf
8. http://sites.computer.org/ccse/SE2004Volume.pdf
9. http://homepages.cs.ncl.ac.uk/brian.randell/NATO/nato1968.PDF
10. http://www.dau.mil/pubs/pdf/SEFGuide 01-01.pdf