

Bansilal Ramnath Agarwal Charitable Trust's
Vishwakarma Institute of Information Technology, Pune-48
(An Autonomous Institute affiliated to Savitribai Phule Pune University)



Curriculum for
T.Y. B.Tech
(Pattern – 2020)

Department of Information Technology



DEPARTMENT OF INFORMATION TECHNOLOGY

• Vision

“To create professionally competent and globally acceptable IT engineers with social awareness”.

• Mission

- Educating budding engineers for, **industry, academia, research and entrepreneurial** pursuit through rigorous implementation of IT curriculum
- Inculcating IT skills to develop **innovative solutions** relevant to **global issues**
- **Imparting values** to practice social and professional ethics.

Program Specific Outcomes (PSOs)

At the end of program, students should be able to

- **PSO a:** An ability to understand, analyze and develop computer programs in the areas related to algorithms, web development and database management
- **PSO b:** An ability to apply knowledge of software engineering principles and practices for multidisciplinary applications to meet the needs of the industry and society

Program Outcomes (POs)

At the end of the program, students should be able to:

1. **Engineering knowledge:** Apply the knowledge of mathematics, science, engineering fundamentals and an engineering specialization to the solution of complex engineering problems.
1. **Problem analysis:** Identify, formulate, review research literature and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences and engineering sciences.
2. **Design/development of solutions:** Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety and the cultural, social and environmental considerations.

3. **Conduct investigations of complex problems:** Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.
4. **Modern tool usage:** Create, select and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
5. **Modern tool usage:** Create, select and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.
6. **The engineer and society:** Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.
7. **Environment and sustainability:** Understand the impact of the professional engineering solutions in societal and environmental contexts and demonstrate the knowledge of, and need for sustainable development.
8. **Ethics:** Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.
9. **Individual and team work:** Function effectively as an individual and as a member or leader in diverse teams and in multidisciplinary settings.
10. **Communication:** Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations and give and receive clear instructions.
11. **Project management and finance:** Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.
12. **Life-long learning:** Recognize the need for and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.

T.Y.B.Tech.
2020 Pattern
Syllabus Structure



T.Y. B. TECH (INFORMATION TECHNOLOGY), SEMESTER V (PATTERN 2020)

Course Code	Course Title	Course Type	Teaching Scheme			Examination Scheme					Total	Credits
			L	T	P	CI E	ISE	SC E	ES E	PR/OR/TW		
ITUA31201	ARTIFICIAL INTELLIGENCE	TH	3	0	2	20	30	20	30	25	125	4
ITUA31202	DATABASE MANAGEMENT SYSTEMS	TH	3	0	2	20	30	20	30	25	125	4
ITUA31203	OPERATING SYSTEMS AND SYSTEM PROGRAMMING	TH	3	1	0	20	30	20	30	25	125	4
ITUA31204	FORMAL LANGUAGE & AUTOMATA THEORY	TH	3	-	-	20	30	20	30	-	100	3
ITUA31205	PROFESSIONAL ELECTIVE-I	TH	3	1	0	20	30	20	30	25	125	4
ITUA31206	PROJECT - I	CE	1	-	2	-	-	-	-	25	25	2
M3	MANDATORY COURSE	AU	-	-	-	-	-	-	-	-	-	-
	Total		16	2	6	100	150	100	150	125	625	21


Professional Elective - I

1. ITUA31205A HUMAN COMPUTER INTERACTION
2. ITUA31205B COMPUTER GRAPHICS
3. ITUA31205C INTERNET OF THINGS

Note- Subjects having only CE components have TW.

Theory: 1Hr. = 1 Credit, Practical: 2 Hrs. = 1 Credit Tut: 1 hr. = 1 Credit, Audit Course: No Credits

Mandatory Courses: Environmental Sciences, Induction training, Indian Constitution, Essence of Indian Traditional Knowledge.


BOS Chairman


Dean Academics


Director





Bansilal Ramnath Agarwal Charitable Trust's
Vishwakarma Institute of Information Technology
 (An Autonomous Institute Affiliated to Savitribai Phule Pune University)

T.Y. B. TECH (INFORMATION TECHNOLOGY), SEMESTER VI (PATTERN 2020)

Course Code	Course Title	Course Type	Teaching Scheme			Examination Scheme					Total	Credits
			L	T	P	CIE	IS E	SC E	ESE	PR/OR / T W		
ITUA32201	DESIGN AND ANALYSIS OF ALGORITHMS	TH	3	1	0	20	30	20	30	25	125	4
ITUA32202	CLOUD COMPUTING	TH	3	0	2	20	30	20	30	25	125	4
ITUA32203	MACHINE LEARNING	TH	3	0	2	20	30	20	30	25	125	4
ITUA32204	PROFESSIONAL ELECTIVE-II	TH	3	1	0	20	30	20	30	25	125	4
	OPEN ELECTIVE (HUMANITIES AND SOCIAL SCIENCE)	TH	3	-	-	20	30	20	30	-	100	3
ITUA32205	PROJECT -II	CE	1	-	2	-	-	-	-	25	25	2
M3	MANDATORY COURSE	AU	-	-	-	-	-	-	-	-	-	-
	Total	-	16	2	6	100	150	100	150	125	625	21

Professional Elective – II

1. ITUA32204A UI/UX DESIGN
2. ITUA32204B MULTIMEDIA AND AR/VR
3. ITUA32204C DIGITAL FORENSICS AND CYBER SECURITY

Note- Subjects having only CE components have TW.

Theory: 1Hr. = 1 Credit, Practical: 2 Hrs. = 1 Credit Tut: 1 hr. = 1 Credit, Audit Course: No Credits

Mandatory Courses: Environmental Sciences, Induction training, Indian Constitution, Essence of Indian Traditional Knowledge.


BOS Chairman


Dean Academics


Director



SEM - I

ARTIFICIAL INTELLIGENCE (ITUA31201)

Teaching Scheme	Examination Scheme						
Credits: 4 Lecture's/Week(L): 3 Hrs/week Practical/Week(P): 2 Hrs/week Tutorial/Week(T): 0 Hrs/week	CIE	ISE	SCE	ESE	PR/OR	TW	TOTAL
	20	30	20	30	25	-	125

Prerequisites: Discrete Mathematics, Data Structures, basic probability theory and statistics, Knowledge of any programming language- (Python, GO)

Course Objectives:

- To understand the various characteristics of intelligent agents and AI domain.
- To learn the different search strategies in AI.
- To learn how to represent knowledge in solving AI problems.
- To introduce the concepts of Expert Systems and Machine Learning.
- To know about the various applications of AI.

Course Outcomes:

After studying this course, students will be able to:

1. Recognize the characteristics of AI agents.
2. Apply various AI search algorithms to typical AI problems.
3. Recognize the importance of knowledge representation and planning.
4. Explain working knowledge of reasoning in presence of incomplete and or uncertain information.
5. Identify different learning techniques to make the system intelligent.
6. Write a paper for in depth exploration of the literature on an aspect of AI application.

Unit I - Introduction to AI and Intelligent Agents

Introduction–Human Intelligence, comparison between Man and Machine, Definition -The Turing Test – Characteristics of Intelligent Agents–Typical Intelligent Agents – Environment, Environment Types, Problem Solving Approach to Typical AI problems, Domains of AI, Brief of Application of AI. Introduction to Basic concept of Machine Learning (ML), Deep Learning (DL) and Data Science (DS). AI vs ML vs DL vs DS.

Case Study- 1 : AI for everyone/ Building AI Projects

Case Study- 2 : AI and Society / Ethics for AI

Unit II – Heuristics Search and Game Playing

Defining the problem as a state space search, problem characteristics, Problem solving Methods - Search Strategies- Uninformed - Informed - Heuristics - Local Search Algorithms and Optimization Problems - Searching with Partial Observations - Constraint Satisfaction Problems – Constraint Propagation - Backtracking Search. (Uninformed Search Strategies: Breadth-first search, Uniform-cost search, Depth-first search, depth-first search, Informed (Heuristic) Search Strategies: Greedy best first search, A* search, Minimizing the total estimated solution cost, Heuristic search techniques- Generate and test, Hill Climbing, Best-First Search, Constraint satisfaction problems (CSP), Game Playing – Alpha Beta Pruning, Minimax search procedure.

Case Study - 3 : State of Art Game Programs and Analysis

Unit III - Knowledge Representation

Knowledge representation, Approach & Issues in knowledge representation, Structured representation of knowledge, Propositional logic, Predicate logic, Introduction to predicate calculus, Resolution, Use of predicate calculus, First order logic, unification and lifting. Weak-slot and filler structure, Strong slot and filler structures.

Unit IV - Knowledge Inference and Planning

<p>Production based system, Frame based system. Inference - Backward chaining, Forward chaining, Rule value approach, Fuzzy reasoning - Certainty factors, Bayesian Theory-Bayesian Network-Dempster – Shafer theory. Planning – Block world problem, components of a planning systems, Goal stack planning, Non-linear planning, Hierarchical planning, least commitment strategy</p>	
<p>Unit V- Learning and Expert System</p>	
<p>Definition of learning, Forms of learning, Rote learning, learning by taking advice, Learning in problem solving, Induction leaning, Explanation based learning, Formal learning theory. Connectionist models- learning in Neural network Expert systems - Intelligent System Vs Expert system, Architecture of expert systems, Roles of expert systems - Knowledge Acquisition –Meta Expert systems shells, Typical expert systems - MYCIN, DART, XOON, RI,.</p>	
<p>Unit VI - AI Applications and Advanced AI</p>	
<p>Natural Language Processing- Introduction, Steps in the process, Spell checking, Language Models – Information Retrieval- Information Extraction - Machine Translation – Speech Recognition , Image formation, Image processing operations, Extracting 3D information, Object Recognition, Using vision_for manipulation and navigation. Robot – Hardware – Perception – Planning – Moving Case Study 4 : Applications of AI</p>	
<p>List Of Assignments</p>	
<ol style="list-style-type: none"> 1. Study Assignment: Compare different algorithms and evaluate their performance/cost. E.g. depth-first search (DFS) to heuristic algorithms such as Monte Carlo tree search (MCTS). Since all of the studied algorithms converge to a solution from a solvable deal, effectiveness of each approach to be measured by how quickly a solution was reached, and how many nodes were traversed until a solution was reached. 2. Assignment on Heuristic Search Techniques: Implement Best--_first search (Best-Solution but not always optimal) 3. Implementation of A* algorithm (Always gives optimal solution) for solving Puzzle problems. 4. Implementation of Min-Max Search Procedure with alpha beta pruning for finding the solutions of games. 5. Assignment on Constraint Satisfaction Problem: Implement graph coloring problem. OR Implementation of Constraint Satisfaction Problem for solving Crypt-arithmetic Problems. Solve any one of the follow: Backtracking: Constraint Satisfaction Problem – 1) Seating Arrangement or Computer CPU utilization 2) Analyze data from twitter or news to categorize them useful reading list out of the whole clutter 3) Read data from images using OCR (optical character recognition) and OMR (optical mark recognition) techniques – use open source libs 6. Implementation of Unification algorithm by considering Resolution concept. 7. Discuss a sample architecture of sensors/actuators which are operated by IA, agent function and characteristics, types of agents (model-based, goal-based...), optimization of agent functions by feedback learning. Support by suitable examples or practical case studies. 8. Mini-project on any one of the following concept (PBL): 1. Expert system 2. Game development 3. NLP or Text Processing 4. Solving problem with AI. 	
<p>Textbooks:</p>	<ol style="list-style-type: none"> 1. Artificial Intelligence: A Modern Approach by Peter and Norvig ISBN-0-13103805- 2. Artificial Intelligence by Elaine Rich, Kevin Knight and Nair ISBN-978-0-07008770-5, TMH 3. Artificial Intelligence by Patrick Henry Winston, Addison-Wesley Publishing Company,ISBN 0-201-53377-4
<p>Reference Books:</p>	<ol style="list-style-type: none"> 1. Prolog Programming for A.I. by Bratko, TMH 2. Artificial Intelligence by Saroj Kausik ISBN:- 978-81-315-1099-5, Cengage Learning

- | | |
|--|--|
| | <ol style="list-style-type: none">3. Artificial Intelligence and Intelligent Systems by Padhy, Oxford University Press4. Deepak Khemani, “A First Course in Artificial Intelligence”, McGraw Hill Education(India), 2013, ISBN : 978-1-25-902998-15. Deepak Khemani —Artificial Intelligence, Tata Mc Graw Hill Education 2013 |
|--|--|

DATABASE MANAGEMENT SYSTEM (ITUA31202)

Teaching Scheme	Examination Scheme						
Credits: 4 Lecture's/Week(L): 3 Hrs/week Practical/Week(P): 2 Hrs/week	CIE	ISE	SCE	ESE	PR/OR	TW	TOTAL
	20	30	20	30	25	-	125

Prerequisites: Discrete Mathematics, Data Structure & Files

Course Objectives:

- To understand the fundamental concepts of database management.
- To provide a strong formal foundation in database concepts, technology.
- To give systematic database design approaches covering conceptual design, logical design and an overview of physical design.
- To learn basic issues of transaction management and concurrency control.
- To learn and understand various Database Architectures and Applications.
- To learn a powerful, flexible and scalable general purpose database to handle big data.

Course Outcomes:

After completion of the course, student will be able to

1. Understand and Design E-R Model for given requirements and convert the same into database tables.
2. Apply database techniques such as SQL & PL/SQL
3. Demonstrate database design concept such as normalization.
4. Demonstrate transaction management techniques in relational database System.
5. Understand different database architectures in real time environment.
6. Apply advanced database Programming concepts in Big Data.

Unit I - Introduction To DBMS

Introduction to Database Management Systems, Advantages of a DBMS over file-processing Systems, Database-System purpose and applications, Levels of Database Systems, Database Languages, Data Models, Components of a DBMS and overall structure of a DBMS, Database Design and ER Model: Entity, Attributes, Relationships, Constraints, types of Keys, Design Process, Entity Relationship Model, ER Diagram, Design Issues, Extended E-R Features, converting E-R & EER diagram into tables.

Unit II - Relational Algebra, SQL And PL/SQL

Introduction to Relational Algebra and Calculus, SQL: Characteristics and advantages, SQL Data Types and Literals, Data independence, DDL, DML, DCL, TCL, SQL Operators, Tables: Creating, Modifying, Deleting, Views: Creating, Dropping, Updating using Views, Indexes, SQL DML Queries: SELECT Query and clauses, Set Operations, Predicates and Joins, Set membership, Tuple Variables, Set comparison, Ordering of Tuples, Aggregate Functions, Nested Queries, Database Modification using SQL Insert, Update and Delete Queries. PL/SQL: Concept of Stored Procedures and Functions, Cursors, Triggers, Roles and Privileges, Embedded SQL, Dynamic SQL.

Unit III – Relational Database Design

Relational Model: Basic concepts, Attributes and Domains, CODD's Rules, Functional Dependencies: Basic concepts, closure of set of functional dependencies, closure of attribute set, canonical cover, Decomposition: lossless join decomposition and dependency preservation, The Process of normalization, 1NF, 2NF, 3NF, BCNF, 4NF, 5NF.

Unit IV - Database Transactions and Query Processing

Basic concept of a Transaction, Transaction Management, Properties of Transactions, Concept of Schedule, Serial Schedule, Serializability: Conflict and View, Testing conflict and view serializability, Recoverable and Non-recoverable Schedules, Concurrency Control: Need, Locking Methods, Deadlocks, Time stamping Methods, Different Crash Recovery methods such as Shadow-Paging and Log-Based Recovery: Deferred and Immediate, Checkpoints, Introduction to Query Processing and Query Optimization.

Unit V – Database Architecture

Introduction to Database Architectures: Multi-user DBMS Architectures, Parallel Databases: Speedup and Scale up, Architectures of Parallel Databases. Distributed Databases: Architecture of Distributed Databases, Distributed Database Design, Distributed Data Storage, Distributed Transaction: Basics, Failure modes, Commit Protocols, Concurrency Control in Distributed Database.

Unit VI - Advances in Databases and Big Data

Introduction to NoSQL, Structured versus Unstructured data, Different NoSQL Data Models, NoSQL using MongoDB, CAP theorem and BASE Properties, Comparative study of SQL and NoSQL, Introduction to Big Data, HADOOP- Building blocks of Hadoop, components of Hadoop-HDFS, MapReduce, HBASE, HIVE.

List of Assignments

1. Design any database with at least 3 entities and relationships between them. Apply DCL and DDL commands. Draw suitable ER/EER diagram for the system.
2. Design and implement a database and apply at least 10 different DML queries for the following task. For a given input string display only those records which match the given pattern or a phrase in the search string. Make use of wild characters and LIKE operator for the same. Make use of Boolean and arithmetic operators wherever necessary.
3. Execute the aggregate functions like count, sum, avg etc. on the suitable database. Make use of built in functions according to the need of the database chosen. Retrieve the data from the database based on time and date functions like now (), date (), day (), time () etc. Use group by and having clauses.
4. Implement nested sub queries. Perform a test for set membership (in, not in), set comparison (<some, >=some, <all etc.) and set cardinality (unique, not unique).
5. Write and execute suitable database triggers. Consider row level and statement level triggers.
6. Write and execute PL/SQL stored procedure and function to perform a suitable task on the database.
7. Write a PL/SQL block to implement all types of cursors.
8. Execute DDL statements which demonstrate the use of views. Try to update the base table using its corresponding view. Also consider restrictions on updatable views and perform view creation from multiple tables.

Student group of size 3 to 4 students should decide the statement and scope of the project which will be refined and validated by the faculty considering number of students in the group.

Draw and normalize the design up to at ER Diagram least 3NF in case of back end as RDBMS. Suggested Directions for development of the mini project.

1. Build a suitable GUI by using forms and placing the controls on it for any application. (E.g Student registration for admission, railway reservation, online ticket booking etc.). Proper data entry validations are expected.
2. Develop two tier architecture and use ODBC/JDBC connections to store and retrieve data from the database. Make a user friendly interface for system interaction. You may consider any applications like employee management system, library management system etc.

Implement the basic CRUD operations and execute a transaction that ensures ACID properties. Make use of commands like commit, save point, and rollback. You may use examples like transfer of money from one account to another, cancellation of e-tickets etc.

<p>Textbooks:</p>	<ol style="list-style-type: none"> 1. Silberschatz A., Korth H., Sudarshan S, Database System Concepts, McGraw Hill Publication, ISBN- 0-07-120413-X, Sixth Edition. 2. Elmasri R., Navathe S., "Fundamentals of Database Systems", 4* Edition, Pearson Education, 2003, ISBN 8129702282. 3. Pramod J. Sadalage and Martin Fowler, —NoSQL Distilled, Addison Wesley, ISBN10: 0321826620, ISBN-13: 978-0321826626 4. “Managing and Using MySQL”, Reese G., Yarger R., King T., Williams H, 2nd Edition, Shroff Publishers and Distributors Pvt. Ltd., ISBN 81 - 7366 - 465 – X 5. MongoDB: The Definitive Guide by Kristina Chodorow
<p>Reference Books:</p>	<ol style="list-style-type: none"> 1. Ramkrishna R., Gehrke J., "Database Management Systems", 3rd Edition, McGraw-Hill, 2003, ISBN 0-07- 123151 –X. 2. C J Date, —An Introduction to Database Systems, Addison-Wesley, ISBN: 0201144719 Connally T., Begg C., "Database Systems", 3rd Edition, Pearson Education, 2002, ISBN 81-7808-861-4 3. MongoDB, O’Reilly Publications. 4. Hadoop, O’Reilly Publications.http://docs.mongodb.org/manual.

OPERATING SYSTEMS AND SYSTEM PROGRAMMING (ITUA31203)

Teaching Scheme	Examination Scheme						
Credits: 4 Lecture's/Week(L): 3 Hrs/week Tutorial/Week(T): 1 Hrs/week	CIE	ISE	SCE	ESE	PR/OR	TW	TOTAL
	20	30	20	30	-	25	125

Prerequisites: Computer Organization and Architecture, Computer Organization and architecture, Processor Architecture and Interfacing, Data Structures, Theory of Computation: DFA, NFA, Regular expressions, Grammars.

Course Objectives:

- To introduce basic concepts and functions of modern operating systems.
- To understand the concept of process, thread management and scheduling.
- To learn the concept of concurrency control.
- To study various Memory and I/O Management techniques.
- To study language processing fundamentals and assemblers.
- To design macro processors.

Course Outcomes:

After completion of the course, student will be able to

1. Analyse the role of Modern Operating Systems.
2. Summarize the concepts of Process Control and Description.
3. Analyse concurrency and locking mechanism in OS.
4. Implement and analyse proper scheduling and memory management policies for the OS.
5. Interpret language processing fundamentals with detail designing of assembler.
6. Describe design of macro processor and working of linker and loader.

Unit I - Overview Of Operating System

Operating System Objectives and Functions, The Evolution of Operating Systems, Developments Leading to Modern Operating Systems, Virtual Machines. BASH Shell scripting: Basic shell commands, shell as a scripting language.

Unit II - Process Description and Control

Process: Concept of a Process, Process States, Process Description, Process Control (Process creation, Waiting for the process/processes, Loading programs into processes and Process Termination), Execution of the Operating System. Threads: Processes and Threads, Concept of Multithreading, Types of Threads, Thread programming Using Pthreads. Scheduling: Types of Scheduling, Scheduling Algorithms, and Thread Scheduling

Unit III - Concurrency Control

Process/thread Synchronization and Mutual Exclusion: Principles of Concurrency, Requirements for Mutual Exclusion, Mutual Exclusion: Hardware Support, Operating System Support (Semaphores and Mutex), Programming Language Support (Monitors). Classical synchronization problems: Readers/Writers Problem, Producer and Consumer problem, Inter-process communication (Pipes, shared memory: system V). Deadlock: Principles of Deadlock, Deadlock Modeling, Strategies to deal with deadlock: The Ostrich Algorithm, Deadlock Prevention, Deadlock Avoidance, Deadlock detection and recovery, An Integrated Deadlock Strategy, Example: Dining Philosophers Problem. Unix Concurrency Mechanism

Unit IV - Memory and I/O Management

Memory Management: Memory Management Requirements, Memory Partitioning: Fixed Partitioning, Dynamic Partitioning, Buddy System, Relocation, Paging, Segmentation. I/O Management and Disk

Scheduling: I/O Devices, Organization of the I/O Function, I/O Buffering, Disk Scheduling (FIFO, SSTF, SCAN, C-SCAN, LOOK, C-LOOK)	
Unit V- Introduction to Systems Programming and Assemblers	
Introduction: Need of System Software, Components of System Software, Language Processing Activities, Fundamentals of Language Processing. Assemblers: Elements of Assembly Language Programming, A simple Assembly Scheme, Pass structure of Assemblers, Design of Two Pass Assembler.	
Unit VI – Macro-Processors, Loaders and Linkers	
Macro Processor: Macro Definition and call, Macro Expansion, Nested Macro Calls and definition, Advanced Macro Facilities, Design of two-pass Macro Processor. Loaders: Loader Schemes, Compile and Go, General Loader Scheme, Absolute Loader Scheme, Subroutine Linkages, Relocation and linking concepts, Self-relocating programs, Relocating Loaders, Direct Linking Loaders, Overlay Structure, Linkers, Dynamic linking and loading.	
List of Tutorial Assignments	
<ol style="list-style-type: none"> 1. Shell programming Write a program to implement an address book with options given below: a) Create address book. b) View address book. c) Insert a record. d) Delete a record. e) Modify a record. f) Exit. 2. The demonstration of FORK, EXECVE and WAIT system calls along with zombie and orphan states. Implement the C program in which main program accepts an integer array. Main program uses the FORK system call to create a new process called a child process. Parent process sorts an integer array and passes the sorted array to child process through the command line arguments of EXECVE system call. The child process uses EXECVE system call to load new program that uses. 3. Thread management using pthread library. Implement matrix multiplication using multithreading. Application should have pthread_create, pthread_join, pthread_exit. In the program, every thread must return the value and must be collected in pthread_join in the main function. Final sum of rowcolumn multiplication must be done by main thread (main function). 4. Inter process communication in Linux using Pipes: Full duplex communication between parent and child processes. Parent process writes a pathname of a file (the contents of the file are desired) on one pipe to be read by child process and child process writes the contents of the file on second pipe to be read by parent process and displays on standard output. 5. Inter-process Communication using Shared Memory using System V. Application to demonstrate: Client and Server Programs in which server process creates a shared memory segment and writes the message to the shared memory segment. Client process reads the message from the shared memory segment and displays it to the screen. 6. Thread synchronization using counting semaphores and mutual exclusion using mutex. Application to demonstrate: producer-consumer problem with counting semaphores and mutex. 7. Reader Writer Problem solution with Mutex , Semaphore. 8. Implement the deadlock-free solution to Dining Philosophers problem to illustrate the problem of deadlock and/or starvation that can occur when many synchronized threads are competing for limited resources. 9. Study Pass I and Pass II of assembler. 10. Study macroprocessor. 	
Textbooks:	<ol style="list-style-type: none"> 1. William Stallings, Operating System: Internals and Design Principles, Prentice Hall, ISBN-10: 0-13- 380591-3, ISBN-13: 978-0-13-380591-8, 8th Edition 2. Abraham Silberschatz, Peter Baer Galvin and Greg Gagne, Operating System Concepts, WILEY, ISBN 978-1-118-06333-0 , 9th Edition 3. D. M. Dhamdhare, Systems Programming and Operating Systems, Tata McGraw Hill, ISBN 13:978-0-07-463579-7, Second Revised Edition 4. Alfred V. Aho, Ravi Sethi, Jeffrey D. Ullman, Compilers Principles, Techniques and Tools, Addison Wesley, ISBN:981–235–885 - 4, Low Price Edition. 5. John R. Levine, Tony Mason & Doug Brown, “Lex&Yacc”, O’Reilly.

Reference Books:	<ol style="list-style-type: none">1. Andrew S. Tanenbaum & Herbert Bos, Modern Operating System, Pearson, ISBN-13: 9780133592221, 4th Edition2. Tom Adelstein and Bill Lubanovic, Linux System Administration, O'Reilly Media, ISBN-10: 0596009526, ISBN-13: 978-0596009526.3. J. J. Donovan, Systems Programming, McGraw-Hill, ISBN 13:978-0-07-460482-3, Indian Edition.
-------------------------	--

FORMAL LANGUAGE & AUTOMATA THEORY (ITUA31204)

Teaching Scheme	Examination Scheme						
Credits: 3 Lecture (L): 3 hrs./week	CIE	ISE	SCE	ESE	PR/OR	TW	Total
	20	30	20	30	-	-	100

Prerequisites: Basic Mathematics, Discrete Structure
Course Objectives : <ul style="list-style-type: none"> ● Understand fundamentals of computer mathematics. ● Study the various abstract computing models. ● Study the Regular expression. ● Study the different types of languages & their relationships. ● Study the different types of grammar & ambiguity in the grammar ● Study the recursive & recursively enumerable languages.
Course Outcomes: After completion of the course, student will be able to <ol style="list-style-type: none"> 1. Use the mathematical preliminaries with the help of proofs and lemmas for language derivation. 2. Understand Regular expression, its conversion to automata and its applications. 3. Understand different types of languages, grammars and removal of ambiguity in the grammar. 4. Construct pushdown automata for formal language and its applications. 5. Understand the concept of language acceptability by Turing machine and types of Turing machine. 6. Understand the recursive & recursively enumerable languages.
Unit I - Theory of Automata
Basic Mathematical Objects: Sets, Logic, functions, Relations, Strings. Definition of Finite Automata (FA), Description of FA, Transition Systems, Acceptability of a String by a FA, Deterministic and Non deterministic FA, Equivalence of DFA and NFA, FA with output: Moore and Mealy machines-Definitions, Models, inter-conversion, minimization of FA. Applications of Finite Automata.
Unit II - Regular Expressions (RE) and Languages
Recursive definition of regular expression, regular set, identities of regular expressions, Conversion of NFA With epsilon moves to DFA, Conversion-RE to DFA , Conversion-DFA to RE, Equivalence of R.E, Equivalence of FA, Arden's theorem, Pumping lemma for regular languages, Closure properties of regular languages, Applications of R.E
Unit III – Grammar
Grammar- Definition, representation of grammar, Chomsky hierarchy, Context Free Grammar- Definition, Derivation, sentential form, parse tree, inference, derivation, parse tree, ambiguity in grammar and language, Simplifications of context free Grammar-Eliminating unit productions, useless symbols, and Null-productions, Normal Forms for CFG- Chomsky normal form.
Regular Grammar- Definition, left linear, right linear grammar, Inter conversion between left linear and right linear regular grammar Application of CFG: Parser, Markup languages
Unit IV - Push Down Automata

Push Down Automata- Definition, Notation, acceptance by final state, acceptance by empty stack, Equivalence of PDA and CFG- Grammar to PDA, PDA to Grammar, Deterministic PDA and Non Deterministic PDA. Closure Properties of CFLs.	
Unit V – Turing machines	
Turing machine Model, Representation of Turing machine, Language acceptability by Turing machine, Design of Turing machine, Types of TM, Halting Problem, Applications of Turing Machine	
Unit VI - Recursively Enumerable Languages	
Recursively Enumerable and Recursive, Properties of recursive and recursively enumerable languages, Post correspondence problem, The Classes P and NP: Problems Solvable in Polynomial Time, An Example: Kruskal's Algorithm, Nondeterministic Polynomial Time, An NP Example: The Traveling Salesman Problem Undecidability: Reducing one problem to another, Other unsolvable Problems. Tractable and In Tractable Problems	
Text books :	<ol style="list-style-type: none"> 1. Mishra K., Chandrasekaran N., "Theory of Computer Science (Automata, Languages and Computation)", Second Edition, Prentice Hall of India 2. John C Martin. "Introduction to Language and Theory of Computation", Third edition, Tata McGraw- Hill.
Reference Books :	<ol style="list-style-type: none"> 1. Hopcroft J., Motwani R., Ullman J., "Introduction to Automata Theory, Languages and Computations", Third edition, Pearson Education Asia.

HUMAN COMPUTER INTERACTION (ITUA31205A)

Teaching Scheme	Examination Scheme						
Credits: 4 Lecture's/Week(L): 3 Hrs/week Practical/Week(P): 0 Hrs/week Tutorial/Week(T): 1 Hrs/week	CIE	ISE	SCE	ESE	PR/OR	TW	TOTAL
	20	30	20	30	-	25	125

Prerequisites: Software Engineering.

Course Objectives:

- Provide an overview of the concepts relating to the design of human-computer interfaces in ways making computer-based systems comprehensive, friendly and usable.
- Understand the theoretical dimensions of human factors involved in the acceptance of computer interfaces.
- Understand the important aspects of implementation of human-computer interfaces.
- Identify the various tools and techniques for interface analysis, design, and evaluation.
- Identify the impact of usable interfaces in the acceptance and performance utilization of information systems.
- Identify the importance of working in teams and the role of each member within an interface development phase.

Course Outcomes:

After studying this course, students will be able to

1. Understand the Human-Computer Interaction and what interaction design is.
2. Understand Human Capabilities and Core Cognitive aspects of interaction design.
3. Design Interactive Systems with clear understanding of HCI principles that influence a system's interface design, before writing any code.
4. Create interfaces for different environments, people, places, and activities.
5. Apply Quantitative Analysis, Evaluation, and Redesign.
6. Evaluate sample interfaces using different models.

Unit I - Introduction to Human-Computer Interaction

Includes the difference between good and poor interaction design, what interaction design is and how it relates to human-computer interaction and other fields, what is involved in the process of interaction design, the different forms of guidance used in interaction design, etc.

Need for Design - Examples from Design of everyday things, case studies, Evolution of the web and digital interfaces, Design thinking and wicked problems.

Exercise - Identify problems around us requiring design solution Or problems solved using design.

Unit II – Understanding the Design Process

Interaction design basics, HCI in the software process, Design rules, Implementation support, Evaluation techniques, Universal design, User support, Individual differences, designing interfaces for all, User research and techniques, Understanding Personae, Good and poor design, Ergonomics.

Exercise - Creating personae for different application in everyday use.

Unit III - Understanding the Interaction

Understanding device specific interactions and human aspects involved, Interaction styles, interacting with voice, visual and audio-visual interfaces, Understanding Scenarios and context of use for user, Understanding user journey and user journey maps.

Interaction Styles- Direct Manipulation and Virtual Environments, Menu Selection, Form Filling and Dialog Boxes, Command and Natural Languages, Interaction Devices, Collaboration and Social Media Participation

Exercise - Listing down scenarios for an application/system, critical task list for each scenario

Exercise - Creating a user journey map for a particular task

Unit IV - Design Issues

Introduction to UX design process and case study, Understanding Information Architecture, Overview of tools. Understanding navigation models based on information architecture, High level concept sketches/wireframes. Quality of Service, Balancing Function and Fashion, User Documentation and Online Help, Information Search, Information Visualization.

Exercise - Open and closed card sorting technique - Creating information architecture for a system

Exercise - Creating low fidelity concept sketches for critical tasks of system/problems.

Unit V - UI Evaluation Techniques

What, why and when to evaluate, Design guidelines, Golden rules and heuristics, Goals of Evaluation, Evaluation criteria, Evaluation through: Expert analysis, User participation, Testing techniques - Formative and Summative testing, surveys, peer reviews and so on.

Case study - ROI on UX/HCI methodology

Unit VI - cognitive models

Cognitive models, Goal and Task hierarchy models, Linguistic models, Physical and Device models, Design principles. Introduction to Prototyping tools, Understanding how UI Interaction & Prototype Design works, UX -Industry overview. Prototyping / wire framing tool exercise: Design a small mobile app design, along with user journeys and multiple interaction touch points using Figma or Balsamiq.

Exercise - Conduct evaluation of different sample interfaces using different models.

List of Assignments

1. Design of Everyday Things: Find 3 unique usability problems in the day-to-day objects that you have observed or encountered. Describe them in 3 slides in a presentation.
2. User Research: Contextual Inquiry: In groups of two, conduct an interview with a newcomer to IIT Bombay in trying to understand their perspective on way-finding in the campus. Record the audio with the user's permission. (If the user disagrees, take notes and prepare transcripts). Try to follow as many sutras of contextual inquiry as possible.
3. One-minute audio clips / transcripts from this interview on a Google classroom. Each clip / transcript should demonstrate your understanding of one of the sutras during the interview. Name the audio file according to the sutra. In an accompanying Word/PDF document, write a short (100 word) description of each audio file / transcript and a justification of the sutra. You may embed photographs to establish the context of the interview. Don't forget to list your names.
4. User Persona: Creating personae for different application in everyday use. Develop a precise description of the user of the product in focus of the design and what the user wishes to accomplish.
5. User journey and User journey maps: Listing down scenarios for an application/system, critical task list for each scenario. Creating a user journey map for a particular task.

Textbooks:	<ol style="list-style-type: none"> 1. “Human Computer Interaction” by Alan Dix, Janet Finlay, ISBN :9788131717035, Pearson Education (2004) 2. “Designing the User Interface - Strategies for Effective Human Computer Interaction”, by Ben Schneiderman, ISBN: 9788131732557, Pearson Education (2010).
Reference Books:	<ol style="list-style-type: none"> 1. Usability Engineering: Scenario-Based Development of Human-Computer Interaction, by Rosson, M. and Carroll, J. (2002) 2. The Essentials of Interaction Design, by Cooper, et al., Wiley Publishing (2007) 3. Usability Engineering, by Nielsen, J. Morgan Kaufmann, San Francisco, 1993. ISBN 0-12-518406-9 4. The Resonant Interface: HCI Foundations for Interaction Design, by Heim, S., Addison-Wesley. (2007) 5. Usability engineering: scenario-based development of human-computer interaction, By Rosson, M.B & Carroll, J.M. , Morgan Kaufman.(2002)

COMPUTER GRAPHICS (ITUA31205B)

Teaching Scheme	Examination Scheme						
Credits: 4 Lecture's/Week(L): 3 Hrs/week Practical/Week(P): - Hrs/week Tutorial/Week(T): 1 Hrs/week	CIE	ISE	SCE	ESE	PR/OR	TW	TOTAL
	20	30	20	30	-	25	125

<p>Prerequisites: Data Structures and algorithms , Basic Mathematics, Geometry, linear algebra, vectors and matrices</p>
<p>Course Objectives:</p> <ul style="list-style-type: none"> ● Gain knowledge about graphics hardware devices and software used. ● Understand the two dimensional graphics and their transformations. ● Understand the three dimensional graphics and their transformations. ● Appreciate animation, illumination and color models. ● Understand OPENGL techniques.
<p>Course Outcomes:</p> <p>After studying this course, students will be able to</p> <ol style="list-style-type: none"> 1. Understand two dimensional graphics. 2. Learn 2D and 3D graphics programming using OpenGL 3. Differentiate and implement various polygon filling/clipping algorithms 4. Study and apply transformations on 2D and 3D objects. 5. Design animation sequences. 6. Distinguish between Illumination and color models.
<p>Unit I – Basic of Computer Graphics</p> <p>Basic of Computer Graphics, Applications of computer graphics, pixel, frame buffer, resolution, aspect ratio. Overview of graphics systems – Video display devices, Raster scan systems, Random scan systems, Graphics monitors and Workstations, Input devices, Hard copy Devices, Graphics Software; Output primitives – points and lines, line drawing algorithms-Digital Differential Analyzer (DDA), Bresenham line algorithms, Line styles: thick, dotted and dashed. Circle drawing algorithm: DDA, Bresenham. Character generating methods: stroke and bitmap method</p>
<p>Unit II - Graphics Programming using OPENGL</p> <p>Why OpenGL, Features in OpenGL, OpenGL operations, Abstractions in OpenGL – GL, GLU & GLUT, 3D viewing pipeline, viewing matrix specifications, a few examples and demos of OpenGL programs</p>
<p>Unit III – Polygons and Clipping Algorithms</p> <p>Introduction to polygon, types: convex, concave and complex. Representation of polygon, Inside test, polygon filling algorithms – flood fill, seed fill, scan line fill and filling with patterns. Windowing and clipping: viewing transformations, 2-D clipping: Cohen – Sutherland algorithm, Polygon clipping: Sutherland Hodgeman algorithm</p>
<p>Unit IV -2D and 3D Transformations</p> <p>Two dimensional viewing – viewing pipeline, viewing coordinate reference frame; widow-to-viewport coordinate transformation, Transformations and Matrices, Transformation Conventions, 2D Transformations, Homogeneous Coordinates and Matrix Representation of 2D Transformations, Translations and Homogeneous Coordinates, Rotation, Reflection, Scaling, Shear transformation Combined Transformation, Transformation of Points, Transformation of The Unit Square, Solid Body Transformations, Rotation About an Arbitrary Point 3-D transformations: introduction, 3-D geometry, primitives, Stages in 3D viewing ,3-D transformations and matrix representation, rotation about an arbitrary axis, 3-D viewing transformations Parallel (Oblique: Cavalier, Cabinet and orthographic: isometric, diametric, trimetric) and Perspective (Vanishing Points – 1 point, 2 point and 3 point)</p>
<p>Unit V – Segment and Animation</p>

Segment: Introduction, Segment table, Segment creation, closing, deleting and renaming, Visibility. Animation: Introduction, Design of animation sequences, Animation languages, Keyframe, Morphing	
Unit VI- Illumination and Shading	
Light sources – basic illumination models – halftone patterns and dithering techniques; Properties of light – Standard primaries and chromaticity diagram; Intuitive colour concepts – RGB colour model – YIQ colour model – CMY colour model – HSV colour model – HLS colour model; Colour selection. Shading Models-Constant, Phong’s model, Gouraud shading	
List of Assignments	
<ol style="list-style-type: none"> 1. Implement basic OpenGL primitives. 2. Implement the DDA algorithm for line drawing. <ol style="list-style-type: none"> a) Implement DDA using Mouse. b) Draw any shape or pattern. c) Draw Dotted, Dashed and Das-Dot-Dash line 3. Implement Bresenham’s algorithm for line drawing. <ol style="list-style-type: none"> a) Draw given pattern. b) Draw Dotted, Dashed and Das-Dot-Dash line 4. Implement Midpoint and Bresenham’s Circle drawing algorithm. 5. Draw the polygons by using the mouse. (Use DDA algorithm for line drawing).Fill the polygon with Boundary fill and Flood fill method. 6. Implement 2D translation, sheer, rotation, reflection and scaling transformations on equilateral triangle or rhombus. 7. Implement Sutherland Hodgman algorithm to clip any given polygon. Provide the vertices of the polygon to be clipped and pattern of clipping interactively. 8. Implement 3D Cube translation, scaling and rotation. <ol style="list-style-type: none"> a) Translations in x, y, z directions b) Rotation by angle 450 about z axis, rotation by 600 about y-axis in succession. c) Scaling in x-direction by a factor of 2, scaling in y- direction by a factor of 3 9. Project based Learning -Animation: Implement any animation. 	
Textbooks:	<ol style="list-style-type: none"> 1. S. Harrington, —Computer GraphicsI, 2nd Edition, McGraw-Hill Publications, 1987, ISBN 0 – 07 – 100472 – 6. 2. D. Rogers, —Procedural Elements for Computer GraphicsI, 2nd Edition, Tata McGraw-Hill Publication, 2001, ISBN 0 – 07 – 047371 – 4. 3. Donald D. Hearn, —Computer Graphics with Open GLI, 4th Edition, ISBN13: 9780136053583.
Reference Books:	<ol style="list-style-type: none"> 1. J. Foley, V. Dam, S. Feiner, J. Hughes, —Computer Graphics Principles and Practicel, 2nd Edition, Pearson Education, 2003, ISBN 81 – 7808 – 038 – 9. 2. D. Rogers, J. Adams, —Mathematical Elements for Computer GraphicsI, 2nd Edition, Tata McGrawHill Publication, 2002, ISBN 0 – 07 – 048677 – 8

INTERNET OF THINGS (ITUA31205C)

Teaching Scheme	Examination Scheme						
Credits: 4	CIE	ISE	SCE	ESE	PR/OR	TW	TOTAL
Lecture's/Week(L): 3 Hrs/week Practical/Week(P): - Hrs/week Tutorial/Week(T): 1 Hrs/week	20	30	20	30	-	25	125

Prerequisites: Data Communication and Networking, Computer Networks

Course Objectives:

- To understand fundamentals of IoT
- To implementing small IoT systems using building blocks of IoT
- To gain knowledge of IoT protocols
- To comprehend fundamentals of security in IoT
- To learn how secure infrastructure for IoT supported by IoT platforms
- To learn real world application scenarios of IoT along with its societal and economic impact using case studies

Course Outcomes:

After studying this course, students will be able to

1. Demonstrate the fundamentals of IoT
2. Understand Things in IoT
3. Apply concepts of IoT to build small IoT Systems
4. Comprehend the IoT protocols
5. Analyze the security issues in IoT and various IoT platforms.
6. Design the real-world applications of IoT

Unit I - Introduction to IOT

IoT: Definition and characteristics of IoT, Internet of Things: Vision, Emerging Trends, Economic Significance, Technical Building Blocks, Physical design of IoT, Things of IoT, IoT Protocols, Logical design of IoT, IoT functional blocks, IoT communication models, IoT Communication APIs, IoT enabling technologies, IoT levels and deployment templates, IoT Issues and Challenges, Applications.

Unit II – IOT Things

Introduction to IoT platforms, Sensors, Actuators, and Smart Objects, Classification of things, Sensor Networks, Connecting Smart Objects: Communications Criteria, IoT Access Technologies: IEEE 802.15.4

Unit III- Protocols for IoT

Existing protocols for M2m(TCP/IP) IoT Protocols Organization, IoT Protocols overview: CoAP, MQTT, IPv6, ZigBee, Bluetooth, Wi-fi, Comparison of Traditional Networking Protocols and IoT Protocols, Issues with IoT Standardization.

Unit IV- IoT & M2M

Machine to Machine, Difference between IoT and M2M, Software define Network, Software define Network for IoT, IoT Physical Devices and Endpoints: Basic building blocks and IoT device, Interoperability in IoT, Introduction to Arduino Programming, Integration of Sensors and Actuators with microcontroller. Programming Raspberry Pi with Python.

Unit V- Microcontrollers and Security in IoT & Cloud

Microcontrollers: Arduino, Raspberry Pi, ESP8266, ESP32 and interfaces.

IoT Security: Vulnerabilities of IoT, Security Requirements, Challenges for Secure IoT, Threat Modelling, Key elements of IoT Security.

Introduction to Cloud Computing, Cloud of Things: Grid/SOA and Cloud Computing, Cloud Middleware, Cloud Standards

Unit VI- IoT Case Studies	
Case Studies: Home Intrusion Detection, Weather Monitoring, System, Air Pollution Monitoring, Smart Irrigation, Smart cities, Health Care	
IOT design affair: Introduction to soldering techniques and its types, Mounting of electronic components on PCB. Introduction and applications of Raspberry pi and Arduino architecture.	
List of Assignments	
<ol style="list-style-type: none"> 1. Study of Raspberry-Pi, Beagle board, Arduino, ESP8266. 2. Study of different operating systems for Raspberry-Pi/Beagle board/Arduino. Understanding the process of OS installation on Raspberry-Pi/Beagle board/Arduino 3. Use Open-source prototype platform- Raspberry-Pi/Beagle board/Arduino and write simple program digital read/write using LED and Switch -Analog read/write using sensor and actuators. LED ON-OFF Alternate LAE Glow Numerical display 4. Use open-source prototype platform- Raspberry-Pi/Beagle board/Arduino and write simple program to read and record the values thorough -Analog read/write using sensor and actuators. 5. Upload data from environmental sensor to cloud server (You can use any public cloud IBM Watson IoT cloud or Google or AWS etc.). 6. Write client applications to get services from the server application using Raspberry Pi/Beagle board. 7. Implement a Mini-project. 	
Textbooks:	<ol style="list-style-type: none"> 1. Internet of Things: A Hands-On Approach Arshdeep Bahga, Vijay Madiseti VPT – Paperback 2015 978- 0996025515 628/- 2. 2. IoT Fundamentals: Networking Technologies, Protocols, and Use Cases for the Internet of Things David Hanes, Gonzalo Salgueiro, Patrick Grossetete Cisco Press – Paperback – 16 Aug 2017 978-1- 58714-456- 1 599.
Reference Books:	<ol style="list-style-type: none"> 1. Smart Internet of things projects Agus Kurniawan Packt - Sep 2016 978-1- 78646- 651-8 2 The Internet of Things Key Olivier Willy Publication 2nd Edition 978 2. Applications and protocols Hersent s 119- 99435-0, 3 The Internet of Things Connecting Objects to the Web Hakima Chaouchi, Willy Publications 978-1- 84821- 140-7.

PROJECT-I (ITUA31206)

Teaching Scheme	Examination Scheme						
Credits: 2 Lecture's/Week(L): 1 Hrs/week Practical/Week(P): 2 Hrs/week	CIE	ISE	SCE	ESE	PR/OR	TW	TOTAL
	-	-	-	-	-	25	25

Prerequisites: Object Oriented Programming, Basic knowledge of database

Course Objectives:

- To learn and understand fundamentals HTML, CSS, JavaScript, GIT
- To understand the object-oriented concepts for designing object-oriented models.
- To understand the use of UML (Unified Modeling Language) for object-oriented analysis and design.
- To describe the step-by-step object-oriented methodology of software development from problem statement through analysis, system design, and class design.
- To understand the issues for implementing object-oriented designs or models.

Course Outcomes:

After studying this course, students will be able to

1. Develop client-side programming using HTML, CSS, and JavaScript. To understand web-based system and various technologies for its implementation.
2. Demonstrate the ability to apply the knowledge of object-oriented analysis and design concepts for solving system modeling and design problems using UML.
3. Design and implement object-oriented models using UML appropriate notations.
4. Design object-oriented structural models by developing class diagrams.
5. Apply the domain and application analysis concept for designing UML Diagrams.

Unit I – Views and Design Modeling using UML

Introduction to Modeling & Unified Approach: Object Oriented Analysis, Object Oriented Design, Modeling Based on UML, Use Case Modeling, Use case scenarios, Class Modeling- Class Associations and Identification of Associations, Generalization/Specialization Relationship, Aggregation and Composition Relationships, Attributes, and Methods Identification. **Activity Diagram:** Activity & Actions, Initial and Final Activity, Activity Partitions, Constraints on Action, Swim Lanes. **Sequence Diagram:** Context, Objects, and Roles, Links, Object Lifeline, Activation/Focus of Control, Modeling Interactions. **State Diagram:** State Machine, Triggers, and Ports, Transitions, Initial and Final State, Composite States, Submachine States. **Component Diagram:** Component designing, input, output, network, process, external & internal components. **Deployment Diagram:** Environment, Access rights, data flow, pipeline & testing trouble shooting & monitoring, Common modeling Techniques: Most frequent methods, Techniques for modeling.

Unit II- Introduction to web-based development

Overview of HTML: Introduction to HTML, Editors, Basic, Elements, Attributes, Headings, Paragraphs, Styles, Formatting, Quotations, Comments, Colors, CSS, Links, Images, Favicon, Tables, Lists.

Introduction to CSS: Introduction, Syntax, Selectors, How To add CSS, Comments, Colors, Backgrounds, Borders, Margins, Padding, Height/Width, Box Model, Outline, Text, Fonts, Icons, Links Lists, Tables, Display, Max-width, Position, Z-index, Overflow, Float, Inline-block, Align, Combinators, Pseudo-class, Pseudo-element, Opacity, Navigation Bar, Dropdowns, Image Gallery, Image Sprites, Attr Selectors, Forms, Counters, Website Layout.

Java Script : Introduction to JS, Where To add JavaScript, Output : Java Script Display possibility, JS Statements, Syntax, Comments, Variables, Introduction to Let keyword , Const, Operators, JS Arithmetic Operations, JS Assignment Operators, Data Types, Functions, Objects, Events, Strings, String Methods, Numbers, Number Methods, Arrays, Array Methods, Dates, Date Formats, Date Get

Methods, Date Set Methods, JS Math Object, Random method, Booleans, JS Comparisons & Logical operators, JS Conditional Statements : if else & else if, Switch Statement, JS loop for- in & of , While loop, JS break & continue , JS Iterables.

List of Assignments

1. Draw an Activity diagram with Swim lanes using UML2.0 Notations for major Use Cases
2. Implement Analysis class Model-class diagram with a suitable object-oriented language
3. Implement the design model with a suitable object-oriented language.
4. Draw a Sequence Diagram for every scenario by using advanced notations using UML2.0
5. Implement the state model with a suitable object-oriented language.
6. 6.Draw a component diagram initialize all possible input, output & internal, external components.
7. Draw Deployment diagram for your problem statement where you must mention each & every step up to deployment.
8. Design web page using HTML, CSS for specified problem statement.
9. Design web page using java script for assignment no 8.
10. Create a version controlling using GIT or GITHUB and add collaborator.

Textbooks:	<ol style="list-style-type: none"> 1. Michael. Blaha, James. Rumbaugh “Object-Oriented Modeling and Design with UML”, 2nd Edition, Pearson Education, 2005. 2. Head First HTML and CSS by Elizabeth Robson and Eric Freeman OREILLY Publications 3. Learning Web Design: A Beginner’s Guide to HTML, CSS, JavaScript, and Web Graphics by Jennifer Robbins 4. Learn HTML, CSS & JavaScript & Build a Website, App & Game – by Young Rewired State
-------------------	---

Reference Books:	<ol style="list-style-type: none"> 1. Ali. Bahrami, “Object Oriented Systems Development”, McGraw-Hill, 2008. 2. Grady. Booch “Object-Oriented Analysis and Design with Applications”, 3rd Edition, Pearson, 2007. 3. Mark. Priestley, “Practical Object-Oriented Design with UML”, 2nd Edition, Tata McGraw-Hill, 2003. 4. HTML and CSS: Design and Build Websites by Jon Duckett 5. HTML, CSS, and JavaScript All in One by Meloni and Kyrin’s pearson publications.
-------------------------	--

SEM - II

DESIGN AND ANALYSIS OF ALGORITHMS (ITUA32201)

Teaching Scheme	Examination Scheme						
Credits: 4 Lecture's/Week(L): 3 Hrs/week Tutorial/Week(T): 1 Hr/week	CIE	ISE	SCE	ESE	PR	TW	TOTAL
	20	30	20	30	-	25	125

Prerequisites: Basic Mathematics, Discrete Structure, Fundamentals of Data structure, Data structures and files.

Course Objectives:

- To understand the problem solving and problem classification.
- To know the basics of computational complexity analysis and various algorithm design strategies.
- To provide students with solid foundations to deal with a wide variety of computational problems.
- To provide a thorough knowledge of the most common algorithms and data structures.
- To analyze a problem and identify the computing requirements appropriate for its solutions.
- To understand basic concepts of P, NP class problems and parallel algorithms.

Course Outcomes:

After completion of the course, student will be able to

1. Calculate computational complexity using asymptotic notations for various algorithms.
2. Apply Divide & Conquer approach to design algorithms.
3. Greedy approach to design algorithms.
4. Practice principle of optimality and Dynamic Programming.
5. Illustrate different problems using Backtracking.
6. Compare different methods of Branch and Bound strategy. Explore the concept of P, NP, NP-complete, NP-Hard.

Unit I - Introduction

Analysis of Algorithm, Efficiency- Analysis framework, asymptotic notations big O, theta and omega. Analysis of Non-recursive and recursive algorithms.

Solving Recurrence Equations : Substitute method and Master Theorem.

Proof Techniques: Minimum 2 examples of each: Contradiction, Mathematical Induction : Tiling Problem, Tower of Hanoi.

Unit II – Divide and Conquer Algorithm

Searching Algorithms: Binary Search, Min-Max Finding and their Analysis

Sorting Algorithms: Merge Sort and Analysis, Quick Sort and Analysis (Best Case, Worst Case and Average Case). Large integer multiplication and Strassen matrix multiplication (Analysis of this algorithm and comparing them with brute force method)

Unit III - Greedy Method

General Method and Characteristics, Knapsack problem, Optimal storage on tapes, Job sequencing with deadlines, Huffman Coding , **Minimum spanning tree** – Prims and Kruskal Algorithm. **Single source shortest path - Dijkstra's algorithm**

Unit IV- Dynamic Programming

Greedy Algorithms vs Dynamic Programming, Recursion vs Dynamic Programming, Memorization Strategy. 0/1 Knapsack Problem, Floyd Warshal Algorithm, Optimal Binary Search Tree, Travelling Salesman Problem, Multistage Graph Problem, longest common subsequence.

Unit V - Backtracking

General method, Recursive backtracking algorithm, iterative backtracking method. 8- queens problem, Sum of subsets, Graph coloring, Hamiltonian Cycle , 0/1 Knapsack Problem

Unit VI - Branch and Bound Method and Computational Complexities	
The method, Control abstractions for Least Cost Search, Bounding, FIFO branch and bound, LC Branch and bound, 0/1 Knapsack problem LC branch and bound and FIFO branch and bound solution. The classes P, NP, NP Complete and NP-hard : Basic concepts, Non-deterministic algorithms, NP-hard graph problems and scheduling problems.	
List of Assignments	
<ol style="list-style-type: none"> 1. Write a program to implement matrix multiplication using Strassen's method. (Divide and Conquer). 2. Implement program to find minimum and maximum element from given list using Divide and Conquer. 3. Write a program to implement optimal storage tape using greedy approach. 4. Write a program to implement longest common subsequence (Dynamic Programming) and verify the complexity. 5. Write a program to print shortest path and cost for the directed graph using Bellman Ford algorithm (Dynamic Programming) and verify the complexity. 6. Write a recursive program to find the solution of placing n queens on chess board so that no queen takes each other (backtracking). 7. Write a non-recursive program to check whether Hamiltonian path exists in undirected graph or not. If exists print it. (Backtracking). 8. Write a program to solve the travelling salesman problem. Print the path and the cost. (Branch and Bound). 	
Textbooks:	<ol style="list-style-type: none"> 1. Horowitz and Sahani, "Fundamentals of computer Algorithms", Galgotia. ISBN 81-7371-612-9.
Reference Books:	<ol style="list-style-type: none"> 1. Thomas H Cormen and Charles E.L Leiserson, "Introduction to Algorithm" PHI, ISBN:81-203-2141-3. 2. AnanyLevitin, "Introduction to the Design & Analysis of Algorithm ",Pearson ISBN 81- 7758-835-4 3. Gilles Brassard, Paul Bratle "Fundamentals of Algorithms ", Pearson ISBN 978-81-317-1244-3 4. George T. Heineman, Gary Pollice, Stanley Selkow "Algorithms in a Nutshell, A Desktop Quick Reference", O'Reilly, ISBN 13:978-81-8404-608-3

CLOUD COMPUTING (ITUA32202)

Teaching Scheme	Examination Scheme						
Credits: 4 Lecture (L): 3 hrs./week Practical (PR): 2hr/week	CIE	ISE	SCE	ESE	PR/OR	TW	Total
	20	30	20	30	25	-	125

Prerequisites: Computer Networks

Course Objectives:

- To understand cloud computing concepts.
- To study supporting technologies of the cloud.
- To study open research problems of cloud computing.
- To study various platforms for cloud computing.
- To explore the applications based on cloud computing.
- To study and evaluate the contemporary technologies in cloud computing.

Course Outcomes:

After completion of the course, students will be able to

1. Summarize the basic concepts of cloud computing (Remember)
2. Explore the supporting technologies of cloud computing (Understand)
3. Analyze the challenges and opportunities in cloud computing (Analyze)
4. Utilize cloud services for the deployment of applications (Create).
5. How technologies link to one another and are used in synchrony (Apply)
6. To explore future trends of cloud computing (Evaluate)

UNIT I - Basics of Cloud Computing

Introduction to Cloud Computing, Overview, Applications, Intranets and the Cloud. Your Organization and Cloud Computing- Benefits, Limitations, Security Concerns. Software as a Service (SaaS) - Understanding the Multitenant Nature of SaaS Solutions, Understanding SOA. Platform as a Service (PaaS)-IT Evolution Leading to the Cloud, Benefits of PaaS Solutions, Disadvantages of PaaS Solutions. Infrastructure as a Service (IaaS)-Understanding IaaS, Improving Performance through Load Balancing, System and Storage Redundancy. Case Study: Google Cloud Platform

UNIT II – Virtualization

Implementation Levels of Virtualization, Virtualization Structures/Tools and Mechanisms, Types of Hypervisors, Virtualization of CPU, Memory, and I/O Devices, Virtual Clusters and Resource Management, Virtualization for Data-Center Automation. Common Standards: The Open Cloud Consortium, Open Virtualization Format. Standards for Security. Case study: Virtual Box, VMware

UNIT III - Data Storage and Security in Cloud

Cloud file systems: GFS and HDFS, Big Table, HBase, and Dynamo Cloud data stores: Data store and Simple DB Cloud Storage-Overview, Cloud Storage Providers, Block storage and object storage. Securing the Cloud- General Security Advantages of Cloud-Based Solutions, Introducing Business Continuity and Disaster Recovery. Disaster Recovery- Understanding the Threats. Case study: Discuss research problems of cloud security. Case study: Firebase

UNIT IV - Amazon Web Services

Services offered by Amazon, **EC2:** Configuring a server, Virtual Amazon Cloud (**VPC**): Subnet, AWS Storage and Content Delivery, Amazon **EBS:** Creating an Elastic Block Store Volume, Adding an EBS Volume to an Instance, Backing up an EBS with snapshot Volume. **S3:** Creating an Amazon bucket and managing associated objects. **Elastic Load Balancer:** Introduction, Creating and Verifying Elastic Load Balancer.

UNIT V - Ubiquitous Clouds and the Internet of Things

Introduction to Ubiquitous computing, Properties of Ubiquitous computing, Smart devices, Cloud Trends in Supporting Ubiquitous Computing, Performance of Distributed Systems and the Cloud, Enabling Technologies for the Internet of Things (RFID and Sensor Networks Technology, GPS), Innovative Applications of the Internet of Things (Smart Homes, Smart Buildings and Smart City), Future Scope of IoT

UNIT VI - Future of Cloud Computing

Docker at a Glance: Process Simplification, Broad Support and Adoption, Architecture, The Docker Workflow, Docker compose file, Docker volume, Docker storage.

Kubernetes: Introduction to Kubernetes, Features of Kubernetes, Kubernetes API, Basic Architecture, Minikube.

Git Hub and DevOps: Introductions and applications

List of Assignments

1. Case study on Google Cloud Platform
2. Write a web-based application and use Firebase.
3. Assignment to study Amazon EC2
4. Assignment to study ELB
5. Creating a Virtual Private Cloud, Subnets, Rout Table, Internet gateway
6. Create a sample web-based application using PHP/Python and deploy it on AWS.
7. Assignment to install and use Docker. Create a Docker file.
8. Assignment to install and use Kubernetes.

Textbooks:

1. Anthony T. Velte Toby J. Velte, Robert Elsenpeter, “Cloud Computing: A Practical Approach”, 2010, the McGraw-Hill.
2. Dr. Kris Jamsa, “Cloud Computing: SaaS, PaaS, IaaS, Virtualization and more”, Wiley Publications, ISBN: 978-0-470-97389-9
3. Gautam Shrof, “ENTERPRISE CLOUD COMPUTING Technology Architecture, Applications, Cambridge University Press, ISBN: 9780511778476
4. Docker Documentation (<https://docs.docker.com/get-started/>)
5. Kubernetes Documentation (<https://kubernetes.io/docs/home/>)

Reference Books:

1. Dr. Kumar Saurabh, “Cloud Computing”, Wiley Publication, ISBN10: 8126536039
2. Buyya, “Mastering Cloud Computing”, Tata McGraw Hill, ISBN-13: 978-1-25-902995-0
3. Barrie Sosinsky, “Cloud Computing”, Wiley India, ISBN: 978-0-470-90356-8
4. Kailash Jayaswal, “Cloud computing”, Black Book, Dreamtech Press
5. Thomas Erl, Zaigham Mahmood and Ricardo Puttini, “Cloud Computing: Concepts, Technology and Architecture”, Pearson, 1st Edition, ISBN :978 9332535923, 9332535922
6. Tim Mather, Subra K, Shahid L., Cloud Security and Privacy, O'Reilly, ISBN-13 978-81-8404-815-5

MACHINE LEARNING (ITUA32203)

Teaching Scheme	Examination Scheme						
Credits: 4 Lecture's/Week(L): 3 Hrs/week Practical/Week(P): 2 Hrs/week Tutorial/Week(T): 0 Hrs/week	CIE	ISE	SCE	ESE	PR/OR	TW	TOTAL
	20	30	20	30	25	-	125

Prerequisites Linear Algebra and Calculus, Probability Basics

Course Objectives:

- Understanding Human learning aspects.
- Understanding primitives and methods in learning process by computer.
- Understanding nature of problems solved with Machine Learning.
- Understanding of building basic machine learning models using Clouds.

Course Outcomes:

After studying this course, students will be able to

1. Comprehend foundation of machine learning and its applied perspective.
2. Understand regression theory.
3. Understand binary and multiclass classification strategies and linear models.
4. Demonstrate and apply algebraic models and logic based models
5. Understand the ANN with its applied perspective.
6. Critically analyze, understand challenges, explore trends and adopt the ML practices to further develop ML based projects.

Unit I - Introduction to Machine Learning

Defining and understanding Machine Learning, Types of machine learning, Model definition, Parametric and non-parametric modeling, Generalization: Concept of Training, Validation, Testing, Overfitting and Under fitting as applied to models. Concept of Bias and Variance and its importance in machine learning. Feature Engineering and Dimensionality reduction. Applications of Machine Learning.

Unit II - Supervised Learning- Regression

Regression: Assessing performance of Regression – Error measures, Linear Models: Least Square method and Polynomial regression, Univariate Regression, Multivariate Linear Regression, Bayesian Linear Regression, Effect of noise in regression.
Principal Component Regression. Error function used in regression
Regularized Regression –overview of Ridge Regression and Lasso

Unit III - Supervised Learning –Classification

Binary and Multiclass Classification: Assessing Classification Performance, Handling more than two classes, Multiclass Classification-One vs One, One vs Rest
Linear Models: Logistic regression, Linear Discriminant Analysis, Bayesian Classification, Support Vector Machines (SVM), Random forest classifiers, Decision Trees Minority Class, Impurity Measures – Gini Index and Entropy, Best Split.

Unit IV - Unsupervised Learning

Principal Components Analysis (Incremental PCA, Sparse PCA, Kernel PCA), Linear Discriminant Analysis, Independent Component Analysis. Distance Based Models: Distance based clustering algorithms - K-means and K-medoids, Hierarchical clustering, Density-based Clustering, Gaussian Mixture models, Maximum Likelihood Estimation. Rule Based Models: Rule learning for subgroup discovery, Association rules mining – Apriori Algorithm, Confidence and Support parameters.

Unit V - Artificial Neural Networks	
Biological neuron, Artificial neuron model, Concept of bias and threshold, Activation functions, McCulloch-Pits Neuron Model, Gradient descent algorithm and application of linear neuron for linear regression and classification, Stochastic Gradient Descent, RMS Prop and Adam optimization techniques. Multilayer perceptron (MLP) and back propagation algorithm, Radial Basis Function networks, Applications of MLP in classification and regression	
Unit VI - Trends in Machine Learning	
Ensemble Learning: Bagging, Randomization, Boosting, Stacking Reinforcement Learning: Exploration, Exploitation, Rewards, Penalties Challenges in Machine learning: Vanishing Gradient problems, Computational Load, Architecture of ConvNet, Convolution Layer, Pooling Layer, and Applications of CNN's in Computer Vision, Introduction to Recurrent neural networks. Case study using CNN.	
List of Assignments	
<ol style="list-style-type: none"> 1. Implement SVM for performing classification and find out the accuracy using given data. 2. Supervised Learning - Regression Generate a proper 2-D data set of N points. Split the data set into Training Data set and Test Data set. i) Perform linear regression analysis with Least Squares Method. ii) Plot the graphs for Training MSE and Test MSE and comment on Curve Fitting and Generalization Error. 3. Create Association Rules for the Market Basket Analysis for the given Threshold. 4. Implement K-means algorithm to create cluster on the given data using python. 5. Implement simple Naïve Bayes algorithm to calculate accuracy, TPR, FPR, TNR, Recall, Precision, error rate etc. measuring parameters on the given data by using Weka API . 6. Implement and test Multi-Layer Perceptron (MLP) trained with back-propagation algorithm. 7. Implement and test Convolutional Neural Network (CNN) for digits recognition. 8. Real time mini project implementation of above mentioned machine learning algorithms for any industry/research oriented problem. 	
Textbooks:	<ol style="list-style-type: none"> 1. EthemAlpaydin: Introduction to Machine Learning, PHI 2nd Edition2013. 2. Peter Flach: Machine Learning: The Art and Science of Algorithms that Make Sense of Data, Cambridge University Press, Edition 2012
Reference Books:	<ol style="list-style-type: none"> 1. C. M. Bishop: Pattern Recognition and Machine Learning, Springer 1st Edition-2013. 2. Ian H Witten, Eibe Frank, Mark A Hall: Data Mining, Practical Machine Learning Tools and Techniques, Elsevier, 3rd Edition. 3. Parag Kulkarni: Reinforcement Learning and Systemic Machine Learning for Decision Making, IEEE Press, Reprint 2015. 4. Nikhil Buduma: Fundamentals of Deep Learning, O'Reilly Media, June 2017. 5. Hastie, Tibshirani, Friedman: Introduction to Statistical Machine Learning with Applications in R, Springer, 2nd Edition 2012. 6. Kevin P Murphy: Machine Learning – A Probabilistic Perspective, MIT Press, August 2012

UI/UX Design (ITUA32204A)

Teaching Scheme	Examination Scheme						
Credits: 4 Lecture / Week (L): 3 Hrs/week	CIE	ISE	SCE	ESE	PR/OR	TW	TOTAL
Practical/Week(P): 0 Hrs/week Tutorials (T): 1 Hrs/week	20	30	20	30	-	25	125

Prerequisites: Human-Computer Interaction, Software Engineering

Course Objectives:

- Understand what user experience (UX) means and how it matters?
- Understand how to approach UX and usability.
- Understand how to approach UI design.

Course Outcomes:

After completion of the course, students will be able to

1. Understand the definition and practical application of user experience
2. Learn multiple User Experience research methods and conduct at least one
3. Create navigation structures
4. Create a medium-fidelity wireframe and a high-fidelity prototype.
5. Analyze research data to guide content, navigation, and design decisions
6. Learn and apply the principles of good interface design

UNIT I - Intro to User Experience Design

Define User Experience: What Does A UX/UI Designer Actually Do? User Experience Process: Universal Principles of User Experience Design – An Introduction, Components of any User Experience Process, Definition of a Good User Experience Design, Components of Good User Experience Design (Usability Heuristics), The 4 Golden Rules of UI Design, Examples of Good User Experience Design, Practical Activity - 1.

UNIT II – User Experience Research

What Is User Experience Research? A Guide to Competitive Analysis for UX Design, Putting Personas to Work in UX Design: What They Are and Why They're Important, User Experience Personas and User Profiles, Other Types of User Experience Research - User Research: Best Practices and Methodologies, Qualitative and Quantitative User Experience Research - Comprehensive Guide to User Experience Research: Methods, Running Interviews, and Observations for User Experience Research - Top UX Research Interview Questions to Ask Users, Understanding User-Testing Data and Results: Translating User Research into User Experience Design, Practical Activity - 2.

UNIT III - Visual Design

Definition of Visual Design: The Role of Visual Design In User Experience, Introduction to Atomic Design: Atomic Design Principles & Methodology, Elements of any Visual Design - Language: Universal Principles of User Experience Design: Language & Typography, Color, and Shape: The Role of Color in Product Design: UX of Color Palettes, Imagery: Effective Use of Images & Graphics in UX Design, Typography in UI Design, Buttons: How to Design and Integrate Buttons Into Your UI, Composition of Visual Design Elements: Key Principles for Better Design Composition, Structure & Grid Layout Grids for Wireframes, Hierarchy of Content- Visual Hierarchy in UX Design, States - Designing States for Buttons & UI Interactions, Examples of Visual Design Elements

across Multiple Platforms & Screen Sizes - Adaptive Design vs. Responsive Design, Practical Activity - 3.

UNIT IV - User Experience Design Strategy

Guide to Information Architecture in UX, Techniques & Best Practice for Developing an Information Architecture - Information Architecture: The Intersection of Users, Content, and Context, Developing an Information Architecture for a Software Product or Solution, Examples of Effective Information Architecture, Developing a Sitemap for a Website - Sitemaps & Information Architecture (IA), Examples for Website Designs, What Is Navigation and Hierarchy?, Examples of Navigation Architecture and Hierarchy of Information, Practical Activity - 4.

UNIT V - Design Mock-Ups

The Fundamentals of Wireframing and Prototyping, User Personas, Scenarios, and Stories: Understanding User Journey vs. User Flow, Design Ideation and Sketching: Product Design Guide: Research, Analysis, Ideation, Storyboarding User Experiences: The What, Why, & When of Storyboarding in UX Design, Design Wireframes, Wireframing: A Blueprint to Your Site, Design Prototyping and Types of Prototypes: Low Fidelity vs. High Fidelity Prototyping, Practical Activity - 5.

UNIT VI - UI/UX Design Tools

Overview of UI/UX Design Tools: Figma, Adobe XD, Sketch

List of Tutorials:

1. Go to your favourite website and explore what is easy to use, and what is difficult.

INSTRUCTIONS:

Using the principles and guidelines learned in Module 1, identify the elements of good user experience design within your selected website, and explain why they are good (or bad). Screen capture the good and bad elements and categorize your findings in Adobe XD. Use Adobe XD to save your screenshots. Divide the canvas into 2 parts. One side for the good UX and one for the bad. Make sure to add a copy explaining why each element is either good or bad. Document your overall findings.

2. You have been tasked with designing an e-commerce mobile app for a local business. Describe how you would approach the research, and which methodologies would be used. Generate a persona or user profile template to be used to summarize your findings.

INSTRUCTIONS:

For this activity make use of brainstorming applications for ideation and documenting research strategies. Get started quickly by using Adobe XD's Whiteboard plugin's easy-to-use templates for persona development, and brainstorming

3. Select any consumer mobile app. Understand the user experience process. Re-design the visual elements of the app by applying the basic principles of visual design.

INSTRUCTIONS:

Adobe XD has many powerful features that help you build the best user experiences, but first you need to learn how to use XD's design, prototype, and sharing capabilities.

4. Using an example of a desktop website of your choice, redesign the navigational experience by creating a sitemap structure (user flow) to ensure the best user experience.

INSTRUCTIONS:

A sitemap is a visual representation of the pages of a website. It often can look like a flow chart diagram of how the pages connect and the path the user would take to navigate the website. In XD it's easy to create a sitemap by using shapes and lines to organize the flow of

the website. You can also use one of the pre-made Whiteboard plugin templates to give you a head start! Whiteboard has several templates that can be used to create your sitemap. <https://spark.adobe.com/page/HWkDHkbd2Jc7s/>

5. Develop a wireframe and a working prototype of a mobile app. Understand the user persona and user stories for the mobile app. Tools such as Adobe XD can be used for this.

INSTRUCTIONS:

Wireframing in XD allows you to focus on the core structure and concept of your interface so you can iterate early in the process, user test, and explore various concepts for the structure of your user interface. Once you've created your wireframe it is easy to add color, graphics, photos, and more to your design to make a real-life high-fidelity prototype. For this activity use the skills you've already mastered in XD to make your mobile app wireframe and high-fidelity mock-up.

Textbooks:	<ol style="list-style-type: none"> 1. Sketching User Experiences: Getting the Design Right and the Right Design, Bill Buxton and William Buxton, 2. A Project Guide to UX Design: For User Experience Designers in the Field or in the Making, Carolyn Chandler and Russ Unger.
Reference Books:	<ol style="list-style-type: none"> 1. Fundamentals of Creating a Great UI/UX, Creative Tim. 2. The Design of Everyday Things, Don Norman 3. UX for Beginners: A Crash Course in 100 Short Lessons, Joel Marsh 4. Designing in Figma, https://figmabook.com

MULTIMEDIA AND AR/VR (ITUA32204B)

Teaching Scheme	Examination Scheme						
Credits: 4 Lecture's/Week(L): 3 Hrs/week Practical/Week(P): NA Tutorial/Week(T): 1 Hrs/week	CIE	ISE	SCE	ESE	PR/OR	TW	TOTAL
	20	30	20	30	-	25	125

Prerequisites: Computer Graphics, Data Base Systems

Course Objectives:

- To learn 5 basic components of multimedia (text, image, audio, video and animation)
- To learn the advance graphics
- To learn compression techniques for various multimedia components
- To learn animation and communication
- To introduce virtual reality concepts

Course Outcomes:

After studying this course, students will be able to

1. Identify the essential features of graphics/image data types, file formats, and colour models in images and video.
2. Critically analyze and apply the major methods and algorithms for multimedia data compression.
3. Demonstrate animation creation, digitized sound processing, video control, and scanned images.
4. Handle multimedia Tools- Hyper Studio, Flash and Photoshop
5. Apply communication in multimedia
6. Understand virtual reality fundamentals.

Unit I – Introduction to Multimedia

Multimedia building blocks, Multimedia architecture, hardware support, Transition from conventional media to digital media. Multimedia Authoring Basics, Some Authoring Tools, Introduction to HyperStudio(Case Study) Multimedia Application: Education. entertainment, edutainment, virtual reality, digital libraries, information kiosks, video on demand, video phone, video conferencing and telemedicine. Usage of text in Multimedia, Families and faces of fonts, outline fonts, bitmap fonts International character sets and hypertext, Digital fonts techniques. Text compression: Huffman coding, LZ & LZW Text file formats: TXT, DOC; RTF, PDF, PS

Unit II - Digital Image

Colour Science, Colour, Colour Models, Colour palettes, Basic Image fundamentals, image File formats - (BMP, TIFF, JPEG, GIF), Basic Image Processing (Case study- Photoshop), Use of image editing software, White balance correction, Dynamic range correction, Gamma correction, Photo Retouching. Image Compression: Types of Compression: Lossy & Lossless, Symmetrical & Asymmetrical, Intra-frame & Inter-frame Hybrid JPEG, Lossless: RLE, Shannon - Fano algorithm, Arithmetic coding. Lossy: Vector quantization, Fractal Compression Technique, Transform Coding, Psycho-analysis, and inter-frame Correlation. Hybrid: JPEG-DCT

Unit III – Audio

Nature of sound waves, characteristics of sound waves, psycho-acoustic, Digitization of sound, frequency and bandwidth, decibel system, data rate, audio file format, Sound synthesis, MIDI, Compression and transmission of audio on Internet, Audio software and hardware. Audio file formats: WAV, AIFF, VOC, AVI, MPEG Audio File formats, RMF, WMA Audio compression techniques: DM, ADPCM and MPEG Audio file conversions Multimedia Supported audio formats in Android, Media Playback.

Unit IV - Video

Video Basics , How Video Works, Broadcast Video Standards, Analog video, Digital video, Video signal formats, Video transmission standards: EDTV, CCIR, CIF, SIF, HDTV, digitization of video Video file formats: MOV, Real Video, H-261, H-263, Cinepack, Nerodigital, Video editing, DVD formats, Video Compression and File Formats. Video compression based on motion compensation, MPEG-1, MPEG-2, MPEG-4, MPEG-7, MPEG-21, Video streaming Multimedia Supported video formats in Android, Media Playback.

Unit V - Animation

Animation: Basics of animation, types of animation, principles of animation, techniques of animation, Creating animation Cell Animation, Key frames, and Twinning ,Morphing and Inverse Kinematics ,Animation Software , Application of Animation , Web-Based Animation , Animation for Learning ,Animation Design and Usage Guidelines

Unit VI - Introduction of Virtual Reality

Introduction of Virtual Reality: Fundamental Concept and Components of Virtual Reality. Primary Features and Present Development on Virtual Reality.

List of Tutorials

1. Study of text compression using Huffman coding.
2. Study of text compression using LZW technique.
3. Study of Shannon Fano algorithm for data compression.
4. Perform Discrete Cosine Transformation.
5. Convert a color image to a negative image.
6. Perform different operations (start, stop, pause, resume etc) on audio clip using Java.
7. Understanding the design a simple MIDI player.
8. Prepare animation sequence on topic of your choice with appropriate animation software.

Textbooks:

1. Tay Vaughan, "Multimedia making it work", Tata McGraw-Hill, 2008.
2. Rajneesh Aggarwal & B. B Tiwari, "Multimedia Systems", Excel Publication, New Delhi, 2007.
3. Li & Drew, "Fundamentals of Multimedia", Pearson Education, 2009..
4. Parekh Ranjan, "Principles of Multimedia", Tata McGraw-Hill, 2007
5. Anirban Mukhopadhyay and Arup Chattopadhyay, "Introduction to Computer Graphics and Multimedia", Second Edition, Vikas Publishing House.
6. Burdea, G. C. and P. Coffet. Virtual Reality Technology, Second Edition. Wiley-IEEE Press, 2003/2006.

Reference Books:

1. Multimedia: Computing, Communications and Applications, Ralf Steinmetz and Klara Nahrstedt, Pearson Education Asia
2. Multimedia Communications, Applications, Networks, Protocols and Standards, Fred Halsall, Pearson Education Asia
3. K.R. Rao, "Multimedia Communication Systems: Techniques, Standards, and Networks", TMH.
4. Ashok Banerji, Ananda Ghosh, "Multimedia Technologies", ISBN: 9780070669239
5. Macromedia Flash 8: A Tutorial Guide, Jay Armstrong, Jen deHaan, BPB Publisher.
6. Photoshop CS4 in Simple Steps, Kogent Learning Solutions Inc., DreamTech Press

DIGITAL FORENSICS AND CYBER SECURITY (ITUA32204C)

Teaching Scheme	Examination Scheme						
Credits: 4	CIE	ISE	SCE	ESE	PR/OR	TW	TOTAL
Lecture's/Week(L): 3 Hrs/week Practical/Week(P): - Hrs/week Tutorial/Week(T): 1 Hrs/week	20	30	20	30	-	25	125

Prerequisites: Digital logic, operating system concepts, Computer hardware knowledge

Course Objectives:

- To understand the basics of digital forensics and techniques for conducting the forensic examination on different digital devices.
- To understand how to examine digital evidence including the data acquisition, identification and analysis.
- To know different types of cyber-attacks and related tools to overcome it

Course Outcomes:

After studying this course, students will be able to

1. Describe Forensic science and Cyber security concepts
2. Determine various modus operandi and motive behind cybercrimes and to identify the digital evidence.
3. Know how to use forensic analysis tools to recover important evidence in crimes involving digital devices.
4. Interpret the significance of evidence, digital forensic process model and their legal perspective.
5. Demonstrate various forensic tools to investigate the cybercrime.
6. Analyze the digital evidence used to commit online offences/frauds.

Unit I - Introduction to forensics and Cyber security

Understanding of forensic science, A brief History of digital and cyber forensics, Forensics , digital forensic, The digital forensic process, Locard's Principle of exchange, Scientific models. Crimes related to digital devices, digital forensics evidence and courts, admissibility in the court of law, legal concerns and privacy related issues. Cyber Crime and information security , Classification of Cyber Crime, Categories of Cyber Crime, Types of Attack

Unit II – Investigation of Digital and Cybercrimes

Understanding Cyber Crime Typologies, Reasons, Causes. Relevant legal sections and acts. Introduction to cybercrime scene, Documenting the scene and evidence, maintaining the chain of custody, forensic cloning of evidence, Live and dead system forensic, Hash value concepts to maintain the integrity of evidence, Report drafting.

Unit III - Data acquisition

Understanding storage formats and storage devices, data acquisition, determining the best acquisition method, acquisition tools, validating data acquisitions, performing RAID data acquisitions, remote network acquisition tools, other forensics acquisitions tools.

Unit IV - Data Recovery

Computing Investigations – Procedure for corporate High-Tech investigations, understanding data recovery work station and software, conducting and investigations

Unit V - Computer Forensics Tools

Evaluating the need of computer Forensics Tools, Types of Computer Forensics Tools, Tasks Performed by Computer Forensics Tools, Tool Comparisons, Computer Forensics Software Tools, Command-Line Forensics Tools, UNIX/Linux Forensics Tools, Other GUI Forensics Tools, Computer Forensics Hardware Tools, Using a Write-Blocker. Basic Malware analysis

Tools and Methods used in Cyber Crime -Proxy server and anonymizers, Phishing password cracking, Key logger and spyware, Types of virus- worms, DoS and DDoS, SQL Injection,

Unit VI - Case Study

Case studies in Social Engineering, Phishing, Vishing, Data Diddling, Salami Attack, Honeytrap, Cryptocurrency frauds, etc.

List of tutorials

1. Study of following tools/e-resources
 - Wireshark
 - COFEE Tool
 - Magnet RAM Capture
 - RAM Capture
 - NFI Defragger
 - Toolsley
 - Volatility
 - FTK
2. Study of different open source forensics platform such as FTK Imager, USB Write blocker, Autopsy etc.
3. Creating a Forensic Image using FTK Imager/Encase Imager :
 - Creating Forensic Image
 - Check Integrity of Data
 - Analyze Forensic Image
4. Perform data acquisition using USB Write Blocker and FTK Imager
5. Solve the Case study (image file) using appropriate standard forensic software (open source tool).
6. Recovering and Inspecting deleted files –
 - Check for Deleted Files
 - Recover the Deleted Files
 - Analyzing and Inspecting the recovered files
7. Email Forensics
 - Mail Service Providers
 - Email protocols
 - Recovering emails
 - Analyzing email header

Textbooks:

1. Warren G. Kruse II and Jay G. Heiser, “Computer Forensics: Incident Response Essentials”, Addison Wesley, 2002.
2. Nelson, B, Phillips, A, Enfinger, F, Stuart, C., “Guide to Computer Forensics and Investigations, 2nd ed., Thomson Course Technology, 2006, ISBN: 0-619-21706-5.
3. The basics of digital Forensics (Latest Edition) – The primer for getting started in digital forensics by John Sammons – Elsevier Syngress Imprint
4. Practical Digital Forensics – Richard Boddington [PACKT] Publication, Open source community
5. Nima Godbole, Sunit Belapure, Cyber Security- Understand Cyber Crimes, Computer Forensics and Legal Perspectives, Wiely India Pvt. Ltd, ISBN- 978-81-265-2179-1.

Reference Books:

1. Vacca, J, Computer Forensics, Computer Crime Scene Investigation, 2nd Ed, Charles River Media, 2005, ISBN: 1-58450-389.
2. Cory Altheide, Harlan Carvey, Digital Forensics with Open Source Tools, Syngress imprint of Elsevier.
3. Bill Nelson, Amelia Phillips, Christopher Steuart, “Guide to Computer Forensics and Investigations”, Fourth Edition, Course Technology.
4. Angus M.Marshall, “Digital forensics: Digital evidence in criminal investigation”, John –Wiley and Sons, 2008.

HUMANITIES AND SOCIAL SCIENCE

Teaching Scheme	Examination Scheme						
	CIE	ISE	SCE	ESE	PR/OR	TW	TOTAL
Credits: 3 Lecture's/Week(L): 3 Hrs/week	20	30	20	30	-	-	100

Prerequisites: NIL

Course Objectives:

- Human and social development.
- Contemporary national and international affairs.
- Emergence of Indian society and Economics.
- Sectoral development and Economic development and related issues (such as international economics, WTO, RBI, etc).

Course Outcomes:

After completion of the course, student will be able to

1. Understand various issues concerning human and society.
2. Realize social, cultural, economic and human issues, involved in social changes
3. Understand the nature of the individual and the relationship between the self and the community
4. Express their opinion about national health and education policies.
5. Understand major ideas, values, beliefs, and experiences that have shaped human history and cultures.
6. Understand the fundamental concepts in engineering economics

Unit I - Indian Society

Structure of Indian Society, Indian Social Demography– Social and Cultural, Differentiations: caste, class, gender and tribe; Institutions of marriage, family and kinship- Secularization –Social Movements and Regionalism- Panchayatraj Institutions; Affirmative Action Programme of the Government-various reservations and commissions.

Unit II - Social Development

Scientific approach to the study of human beings. Evolution of human kind, social change and evolution. Industrial revolution. National policy on education, health and health care and human development.

Unit III – Sectoral Development

Agriculture: Technology changes, Green revolutions, Employment Rural and Urban, Government Schemes.

Industrial Development: Strategies, Public and Private Sectors, Categories, infrastructure, transport and communication, Consumer Awareness.

Unit IV - Economic Development

Need for planned economic development – Law of demand and supply. Planning objective, five years plan, priorities and problems. Population and development.

Indian Economics – basic features, natural resources population size and composition, national income concepts, micro economics of India, inflation, GDP.

Unit V - Banking and Trades

Financial Analysis, Ratios, Cost Analysis, financial Institutions, Finance Commissions, Budget Analysis. Indian Banking, Role of Reserve bank of India International Economy, WTO, International aid for economic growth.

Unit VI - Understanding Cash Flow and Taxes

Accounting for Depreciation and Income Taxes, Project Cash-Flow Analysis, Understanding Financial Statements, Case Studies - cash flow analysis done in start-up companies.

Textbooks:	<ol style="list-style-type: none">1. Krugman, International Economics, Pearson Education.2. Prakash, The Indian Economy, Pearson Education.3. Thursen Gerald, Engineering Economics, Prentice Hall.4. C.S. Rao, Environmental Pollution Control Engineering, New Age International Pvt. Ltd.
Reference Books:	<ol style="list-style-type: none">1. Rangarajan, Environmental Issues in India, Pearson Education.2. University of Delhi, The Individual & Society, Pearson Education.3. Wikipedia.org / wiki /social studies.4. M. N. Srinivas, Social change in modern India, 1991, Orient Longman.5. David Mandelbaum, Society in India, 1990, Popular

PROJECT-II (MODULE I - ANDROID DEVELOPMENT) (ITUA32205)

Teaching Scheme	Examination Scheme						
	CIE	ISE	SCE	ESE	PR/OR	TW	TOTAL
Credits: 2							
Lecture's/Week(L): 1 Hrs/week							
Practical/Week(P): 2 Hrs/week	-	-	-	-	-	25	25

Prerequisites: Fundamentals of Data Structures, Object Oriented Programming, Data Structures and Files, Java Programming

Course Objectives:

- To learn how to employ Integrated Development Environment(IDE) for implementing and testing of software solution in android development
- To acquire contemporary skills to make the students employable in information technology domain

Course Outcomes:

After studying this course, students will be able to:

1. Understand Android activities life cycle.
2. Demonstrate operations on GUI objects.
3. Perform & evaluate Event driven programming.
4. Apply various techniques on working with menu.

Unit I - Introduction to Android, GUI Design Concepts and Android Activities

Overview of Android: What does Android run On – Android Internals? Android for mobile apps development Environment setup for Android apps Development Framework - Android- SDK,

Eclipse Emulators – What is an emulator Android AVD?

Android Emulation – Creation and set up First Android Application.

Design criteria for Android Application : Hardware Design Consideration, Design Demands For Android application, Intent, Activity, Activity Lifecycle and Manifest Creating Application and new Activities Simple UI -Layouts and **Layout properties** :Introduction to Android UI Design, Introducing Layouts XML

Introduction to GUI objects viz.: Push Button, Text / Labels, Edit Text, Toggle Button, Padding.

Unit II – Advanced UI Programming: Toast, Menu, Dialog, List and Adapters

Event driven Programming in Android (Text Edit, Button clicked etc.) **Menu:** Basics, Custom v/s System Menus, Create and Use Handset menu Button (Hardware) **Dialog** : Creating and Altering Dialogs , **Toast** : List & Adapters

Demo Application Development and Launching Basic operation of SQLite Database 5.6 Android Application Priorities

List of Assignments

Prerequisites Software: Java development kit(JDK), Android SDK, Eclipse IDE/Android Studio and Android development tools (ADT) plugins

1. Create application for demonstration of android activity life cycle: Create Registration page to demonstration of Basic widgets available in android.
2. Create sample application with login module.(Check username and password) On successful login, Change Text View “Login Successful” and if login fails display alert user using Toast “Login fail”, And Validate the username and Password(Login button should remain disabled till validation is right).
3. Create an application for demonstration of Scroll view in android: Create an application for demonstration of Explicitly Starting New Activity using Intent.
4. Create an application that will pass two number using TextView to the next screen, and on the next screen display sum of that number.
5. Create spinner with strings taken from resource folder (res >> value folder). On changing spinner value, change background of screen.

	<ol style="list-style-type: none"> 6. Create an application that will Demonstrate Button onClick() Event and change the TextView Color based on button Clicked. 7. Create an UI such that, one screen have list of all the types of cars. On selecting of any car name, next screen should show Car details like: name, launched date, company name. 8. Create an application that will Demonstrate Dialog Box Control In Android 4
Textbooks:	<ol style="list-style-type: none"> 1. Building Android Apps IN EASY STEPS McGraw-Hill Education 2. Professional Android 2 Application Development Reto Meier Wiley India Pvt Ltd 3. Beginning Android Mark L Murphy Wiley India Pvt Ltd 4. Pro Android Sayed Y Hashimi and Satya Komatineni Wiley India Pvt Ltd
Reference Books:	<ol style="list-style-type: none"> 1. Android Studio Development Essentials by Neil Smyth 2. The Definitive Guide to SQL Lite by Michael Owens
Web References:	<ol style="list-style-type: none"> 1. http://www.tutorialspoint.com/android/ http://www.tutorialspoint.com/android/android_overview.htm 2. http://www.codelearn.org/android-tutorial/android-introduction 3. http://pl.cs.jhu.edu/oose/resources/android/Android-Tutorial.pdf 4. http://mobisys.in/blog/2012/01/introduction-to-android-sqlite-database/ 5. www.appmakr.com/Android 6. www.telerik.com/android-development 7. developer.android.com/training/basics/firstapp

Project-II (MODULE II - MERN STACK) (ITUA32205)

Teaching Scheme	Examination Scheme						
Credits: 2	CIE	ISE	SCE	ESE	PR/OR	TW	TOTAL
Lecture's/Week(L): 1 Hrs/week Practical/Week(P): 2 Hrs/week	-	-	-	-	-	25	25

Prerequisites: Data Structure, HTML, CSS, Javascript, Bootstrap And Mongodb

Course Objectives:

- Student will learn about backend and front end.
- Student will learn NodeJS and ExpressJS
- Student will learn how backend is connected with database.
- Student will learn about full stack development.

Course Outcomes:

After studying this course, students will be able to

1. Create a backend server side application.
2. Can create a connection with database.
3. Create front-end Application.
4. Can create full stack application.

Unit I - Introduction to ES6 Syntax, Nodejs, Expressjs And Mongodb

Introduction to ES6 Syntax - The let keyword, The const keyword, Arrow Functions, For/of, Map Objects, Set Objects, Classes, Promises, Symbol, Default Parameters, Function Rest Parameter, String.includes(), String.startsWith(), String.endsWith(), Array.from(), Array keys(), Array find(), Array findIndex(), Array include function, lambda function, async and await, Object Destructuring, Promises, Math Methods, New Number Properties, New Number Methods, New Global Methods, Object entries, ESM Import Export Modules.

NodeJS – Introduction to Node.js, Modules, HTTP Module, File System, URL Module, NPM, Events, Upload Files, Email.

MongoDb - Create Database, Create Collection, Insert, Find, Query, Sort, Delete, Drop Collection, Update, Limit, Join.

ExpressJS and connectivity with MongoDB - Introduction, Routing, HTTP Methods, URL Building, Middleware, Templating, Static Files, Form Data, Database, Mongoose library functions, Cookies, Sessions, Authentication, RESTful APIs, Scaffolding, Error handling, - Debugging, Best Practices

Unit II – ReactJS

React JS - Introduction, Installation, Architecture, Creating a React Application, JSX, Components, Styling , Properties (props) , Event management , State Management , Http client programming , Form programming , Routing , Redux , Animation , Testing , CLI Commands.

Authorization and Authentication, Unit Testing, Building and Deployment on Heroku

List of Assignments

Prerequisites Software: NodeJS and MongoDB

1. Library management using ES6 Syntax which should include all array function, Lambda Functions, Classes and Object, Spread operator and Object Destructuring.
2. Create a HTTP server using NodeJS and ExpressJS – Create a calculator app which will be HTML page, and will serve using NodeJS and ExpressJS
3. CRUD operation using expressJS (Expense Table)
4. Front End using ReactJS (Expense Management)
5. ExpressJS, Mongoose in ReactJS – Front End (Expense Management)
6. Connect with Back End (Expense Management)
7. Authentication and Authorization Back End (Expense Management)
8. Authentication and Authorization Front End (Expense Management)

9. Create a Mini Project and Host on Heroku

Textbooks:	<ol style="list-style-type: none"> 1. Vasan Subramanian, “Pro MERN Stack: Full Stack Web App Development with Mongo, Express, React, and Node”, Publisher(s): Apress, May 2019. ISBN: 9781484243916 2. Eddy Wilson Iriarte Koroliova, “MERN Quick Start Guide”, Publisher(s): Packt Publishing, May 2018. ISBN: 9781787281080
Reference Books:	<ol style="list-style-type: none"> 1. Shama Hoque, “Full-Stack React Projects: Learn MERN stack development by building modern web apps using MongoDB, Express, React, and Node.js, 2nd Edition”, Publisher(s): Packt Publishing. ISBN-13: 978-1839215414
Web References:	<ol style="list-style-type: none"> 1. https://nodejs.dev/learn 2. https://expressjs.com/ 3. https://reactjs.org/tutorial/tutorial.html 4. https://www.tutorialspoint.com/nodejs/index.htm 5. https://www.w3schools.com/nodejs/ 6. https://www.tutorialspoint.com/expressjs/index.htm 7. https://www.w3schools.com/REACT/DEFAULT.ASP 8. https://www.javatpoint.com/reactjs-tutorial 9. https://www.tutorialspoint.com/reactjs/index.htm 10. https://dev.to/tienbku/authentication-authorization-with-react-js-example-18e