Scheme of Instruction and Syllabi
of
Choice Based Credit System (CBCS) of

BE / B.TECH V AND VI SEMESTERS
OF
FOUR YEAR DEGREE COURSE
IN

INFORMATION TECHNOLOGY

CHAITANYA BHARATHI INSTITUTE OF TECHNOLOGY™
(An Autonomous Institution)
Affiliated to OU; All U.G. and 5 P.G. Programmes (Civil, CSE, ECE, Mech. & EEE)
Accredited by NBA; Accredited by NAAC - 'A' Grade (UGC); ISO Certified 9001:2015
Chaitanya Bharathi P.O, CBIT Campus, Gandipet, Kokapet (V),
Gandipet Mandal, Ranga Reddy District, Hyderabad-500075, Telangana
email: principal@cbit.ac.in; Website: www.cbit.ac.in Ph : 040-24193276 / 277 / 279

CHAITANYA BHARATHI INSTITUTE OF TECHNOLOGY(A)
Choice Based Credit System (with effect from 2018-19)
B.E (Information Technology)
SEMESTER – V

S.No	Course Code	Title of the Course	Scheme of Instruction		Scheme of Examination			Credits
			Hours per week		Duration of SEE in Hours	Maximum Marks		
			L/T	P/D		CIE	SEE	
THEORY								
1	16ITC16	Principles of Operating Systems	3	-	3	30	70	3
2	16ITC17	Database Systems	3/1	-	3	30	70	4
3	16ITC18	Software Engineering	3	-	3	30	70	3
4	16ITC19	Web Technology	3	-	3	30	70	3
5	16ITC20	Theory of Automata	3/1	-	3	30	70	4
		Elective - I	3	-	3	30	70	3
PRACTICALS								
6	16ITC21	Operating Systems and Web Technology Lab	-	3	3	25	50	2
7	16ITC22	Database Systems Lab	-	3	3	25	50	2
8	16ITC23	Mini Project-III	-	2	-	50	-	1
TOTAL			**20**	**8**	**-**	**280**	**520**	**25**

L: Lecture T: Tutorial D: Drawing P: Practical
CIE - Continuous Internal Evaluation SEE - Semester End Examination

Elective-I

S.No.	Subject Code	Subject Name
1.	16ITE01	Python Programming
2.	16ITE02	UNIX and Shell Programming
3.	16ITE03	Scripting Languages

16ITC16

PRINCIPLES OF OPERATING SYSTEMS

Instruction	3L Hours per week
Duration of End Examination	3 Hours
Semester End Examination	70 Marks
CIE	30 Marks
Credits	3

Course Objectives:

This course is introduced to

1. Learn various services provided by an operating system.
2. Learn, what a process is and how processes are synchronized and scheduled.
3. Learn different approaches of memory management.
4. Familiarizewith the structure and organization of the file system.
5. Familiarize with Protection and security aspects of operating systems.

Course Outcomes:

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

1. Understand the services of an operating system, inter process communication and multithreaded programming.
2. Identify suitable process scheduling, deadlocks handling algorithms and solve process-synchronization problems.
3. Understand the organization of Main and Virtual memory in the operating system.
4. Understand File-System management.
5. Understand the Security problems, Threats and Protection mechanisms.
6. Choose an efficient algorithm based on different aspects for better performance of the system.

Prerequisites:

Computer Organization and Microprocessor (16ITC11), Programming and Problem Solving (16CSC01), Data Structures & Algorithms (16ITC02).

UNIT–I

Introduction: Definition of Operating System, Computer-System Organization, Computer-System Architecture, Operating-System Structure, Operating-System Operations, Process Management, Memory Management, Storage Management, Protection and Security Kernel Data Structures Computing Environments , Open-Source Operating Systems .

Operating System Structures: Operating-System Services, User and Operating-System InterfaceSystem Calls, Types of System Calls, System Programs, Operating-System Design and Implementation, Operating-System Structure, System Boot.
Process: Process Concept, Process Scheduling, Operations on Processes, Interprocess Communication.
Threads: Overview, Multicore Programming, Multithreading Models, Threading Issues.

UNIT–II
Process Synchronization: Background, The Critical-Section Problem, Peterson's Solution, Synchronization Hardware, Mutex Locks, Semaphores, Classic Problems of Synchronization, Monitors.
CPU Scheduling: Basic Concepts, Scheduling Criteria, Scheduling Algorithms, Thread Scheduling, Multiple-Processor Scheduling, Algorithm Evaluation.
Deadlocks: System Model, Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery from Deadlock.

UNIT–III
Memory: Background, Swapping, Contiguous Memory Allocation, Segmentation, Paging, Structure of the Page Table.
Virtual Memory: Background, Demand Paging, Copy-on-Write, Page Replacement, Allocation of Frames, Thrashing, Memory-Mapped Files, Allocating Kernel Memory.

UNIT–IV
Mass-Storage Structure: Overview of Mass-Storage Structure, Disk Structure, Disk Attachment, Disk Scheduling, Disk Management, Swap-Space Management
File-System Interface: File Concept, Access Methods, Directory and Disk Structure, File-System Mounting, File Sharing Protection.
File-System Implementation: File-System Structure, File-System Implementation, Directory Implementation, Allocation Methods, Free-Space Management, Efficiency and Performance.

UNIT–V
Protection: Goals of Protection, Principles of Protection, Domain of Protection Access Matrix, Implementation of the Access Matrix, Access Control, Revocation of Access Rights, Capability-Based Systems.

Security: The Security Problem, Program Threats, System and Network Threats, Cryptography as a Security Tool, User Authentication.

Text Book:

1. Abraham Silberschatz, Peter Galvin, Greg Gagne, "Operating System Concepts", Ninth Edition, John Wiley and sons publications, 2013.

Suggested Reading:

1. A.Tanenbaum, "Modern Operation Systems", Third Edition, Pearson Education, 2008.
2. William Stallings, "Operating Systems", Fifth Edition, Pearson Education, 2005.
3. Ida M.Flynn, "Understanding Operating Systems", Sixth Edition, Cengage, 2011.
4. D.M.Dhamdhere,"Operating systems a concept based approach", SecondEdition, McGraw-Hill, 2007.
5. Pramod Chandra P.Bhatt, "An Intoduction to Opearting Systems concepts and practice", Third Edition, PHI, 2014.

Web Resources:

1. http://nptel.ac.in/downloads/106108101/
2. http://www2.cs.uic.edu/~jbell/CourseNotes/OperatingSystems/
3. http://www.cs.kent.edu/~farrell/osf03/oldnotes/

16ITC17

DATABASE SYSTEMS

Instruction	3L+1T Hours per week
Duration of End Examination	3 Hours
Semester End Examination	70 Marks
CIE	30 Marks
Credits	4

Course Objectives:

This course is introduced to

1. Familiarize with the fundamental concepts and the role of a database system in an organization.
2. Acquire knowledge on different issues in the design and implementation of a database system.
3. Learn how to write simple and moderately advanced database queries using SQL.
4. Learn logical database design and various database models.
5. Study the concepts of database security, concurrency and recoverability.

Course Outcomes:

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

1. Understand the purpose of database systems and Design any domain specific database using E-R model by considering all the constraints and issues in the related domain.
2. Design and implement a database for any specified domain according to the well-known Relational data model and formulate Relational algebra expressions.
3. Use SQL for efficient data retrieval queries, advanced SQL concepts to access databases from programming languages and define various triggers to ensure the consistency of the databases.
4. Understand and apply normalization concepts in the design of a relational database.
5. Efficiently organize and manage data using indexing and hashing concepts to achieve good data retrieval performance.
6. Understand the concept of a database transaction and related database facilities, including concurrency control, backup and recovery, and data object locking and protocols**.**

Prerequisites:

Data Structuresand Algorithms (16ITC02), Java programming (16ITC10)

UNIT-I

Introduction: Database-System Applications, Purpose of Database Systems, View of Data, Database Languages, Relational Databases, Database Design, Data Storage and Querying, Transaction Management, Database Architecture, Data Mining and Information Retrieval Specialty Databases, Database Users and Administrators.

Database Design and the E-R Model: Overview of the Design Process, the Entity-Relationship Model, Constraints, Removing Redundant Attributes in Entity Sets, Entity-Relationship Diagrams, Reduction to Relational Schemas, Entity-Relationship Design Issues, Extended E-R Features, Alternative Notations for Modeling Data.

UNIT–II

Introduction to the Relational Model: Structure of Relational Databases, Database Schema, Keys, Schema Diagrams, Relational Query Languages, Relational Operations.

Introduction to SQL: Overview of the SQL Query Language, SQL Data Definition, Basic Structure of SQL Queries, Additional Basic Operations, Set Operations, Null Values, Aggregate Functions, Nested Sub queries, Modification of the Database.

Intermediate SQL: Join Expressions, Views, Transactions, Integrity Constraints, SQL Data Types and Schemas, Authorization.

UNIT–III

Advanced SQL: Accessing SQL from a Programming Language, Functions and Procedures, Triggers, Recursive Queries, Advanced Aggregation Features.

Relational Database Design: Features of Good Relational, Designs, Atomic Domains and First Normal Form, Decomposition using Functional Dependencies, Functional-Dependency Theory, Algorithms for Decomposition, Decomposition using Multivalued Dependencies.

UNIT-IV

Indexing and Hashing: Basic Concepts, Ordered Indices, B+ Tree Index Files, Multiple-Key Access, Static Hashing, Dynamic Hashing, Comparison of Ordered Indexing and Hashing, Bitmap Indices, Index Definition in SQL

Transactions: Transaction Concept, a Simple Transaction Model, Transaction Atomicity and Durability, Transaction Isolation, Serializability, Transaction Isolation and Atomicity, Transaction Isolation Levels.

UNIT–V

Concurrency Control: Lock-Based Protocols, Deadlock Handling, Multiple Granularity, Timestamp-Based Protocols, Validation-Based Protocols, Multiversion

Schemes, Snapshot Isolation, Insert Operations, Delete Operations and Predicate Reads.

Recovery System: Failure Classification, Storage, Recovery and Atomicity, Recovery Algorithm, Buffer Management, Failure with Loss of Nonvolatile Storage, ARIES, Remote Backup Systems.

Text Book:

1. Abraham Silberschatz, Henry F Korth, S. Sudarshan, "Database System Concepts", Sixth Edition, McGraw-Hill International Edition, 2010.

Suggested Reading:

1. C J Date, "An Introduction to Database Systems", Eighth Edition, Pearson Education, 2003.
2. RamezElmasri, Shamkant B. Navathe, "Fundamentals of Database System", Sixth Edition, Addison-Wesley, 2011.
3. Raghu Ramakrishnan,JohannesGehrke, "Database Management Systems", Third Edition, McGraw-Hill International Edition, 2014.
4. Patric O'Neil, Elizabeth O'Neil, "Database-principles, programming and performance", Second edition, Morgan Kaufmann Publishers, 2001.

Web Resources:

1. http://db-book.com/
2. https://www.tutorialspoint.com/dbms/
3. https://www.w3schools.in/dbms/
4. http://www.oracle-dba-online.com/sql/oracle_sql_tutorial.htm.
5. http://www.tutorialspoint.com/plsql/

16ITC18

SOFTWARE ENGINEERING

Instruction	3LHours per week
Duration of End Examination	3 Hours
Semester End Examination	70 Marks
CIE	30 Marks
Credits	3

Course Objectives:

This course is introduced to

1. Describe the various software life cycle models.
2. Explain the importance of the software development process.
3. Acquaint the students with software requirements and SRS document.
4. Familiarize the students with different software architectural styles.
5. Explain the importance of software quality and review techniques.

Course Outcomes:

Upon successful completion of this course, the students should be able to

1. Understand the nature of software and definition of software engineering, agile software development and agile process models.
2. Recognize the minimum requirements for the development of application.
3. Develop a system, component, or process to meet desired needs of a customer, conduct tests using various testing methods to verify and validate the results.
4. Involve in developing, maintain, efficient, reliable and cost effective software solutions.
5. Understand the risks, formulate and implement software projects.
6. Use the techniques, skills, and modern engineering tools necessary for engineering practice.

Prerequisites:

Programming and Problem Solving (16CSC01), Design and analysis of algorithms (16ITC08).

UNIT-I

Software and Software Engineering: The Nature of Software, Software Engineering. The Software Process, Software Engineering Practice.

A Generic view of Process : Software Engineering -A Layered Technology, A Process frame work, Process Models-Waterfall model, spiral model, The Unified Process, Product and Process, Process Assessment and Improvement, The CMMI,

Agility: Introduction to Agile development, Product development in Internet time, Agile Process models-Scrum, Extreme programming, Agile Vs Waterfall Model.
Understanding Requirements: Requirements Engineering, Establishing the Groundwork, Eliciting Requirements, Developing Use Cases, Building the Requirements Model, Negotiating Requirements, Validating Requirements.
Requirements Modelling: Requirements Analysis, Scenario-Based Modeling, Problem Analysis, Data Flow Diagrams, Software Requirement and specifications, Behavioural and non-behavioural requirements.

UNIT-II
Design Concepts: Design within the Context of Software Engineering, The Design Process, Design Concepts. Cohesion & Coupling, Object Oriented Design-Identifying Objects and classes, User Interface Design.
Architectural Design: Software Architecture, Architecture Styles-pipe and filter architecture, black board architecture , layered architecture.
Component level Design: Designing Class Based Components, Conducting Component-Level Design, Designing Traditional Components, Component-Based Development.

UNIT-III
Quality Concepts: Software Quality, Achieving Software Quality.
Review Techniques: Cost Impact of Software Defects.
Software Quality Assurance: Background Issues, Elements of Software Quality Assurance, SQA Tasks, Goals and Metrics, Formal Approaches to SQA, Statistical Software Quality Assurance, Software Reliability, The ISO 9000 Quality Standards, The SQA Plan.

UNIT-IV
Software Testing Strategies: A Strategic Approach to Software Testing, Strategic Issues, Validation Testing, System Testing, The Art of Debugging. Testing Tools – Rational functional tester, Testing Standards, Selenium software testing tool.
Testing Conventional Applications: Software Testing Fundamentals, Internal and External Views of Testing, White-Box Testing, Basis Path Testing, Control Structure Testing, Black–Box Testing, system testing, validation testing, beta testing, alpha testing, acceptance testing, regression testing,
Product Metrics: A Framework for Product Metrics, Size Metrics like LOC, Function points.

UNIT-V

Estimation: Observations on Estimation, the Project Planning Process, Software Scope and Feasibility, Resources, Software Project Estimation.

Risk Management: Reactive versus Proactive Risk Strategies, Software Risks, Risk Identification, Risk Projection, Risk Refinement, Risk Mitigation, Monitoring, and Management, The RMMM Plan.

Text Books:

1. Roger S.Pressman, "Software Engineering: A Practitioners Approach", 7th edition, McGrawHill, 2009.
2. Jim Highsmith, "Agile Software Development Ecosystems", Addison-Wesley 2002, ISBN 0201760436, 2010.

Suggested Reading:

1. Carlo Ghezzi, Mehdi Jazayeri, Dino Mandrioli, "Fundamentals of Software Engineering", PHI, 2nd edition,2010.
2. Ali Behforoz and Frederic J.Hadson, "Software Engineering Fundamentals", Oxford End Press, 2010.
3. Pankaj Jalote, "An Integrated Approach to Software Engineering", 3rdedition, Narosa Publishing house, 2008.
4. James F.Peters, WitoldPedrycz, "Software Engineering-An engineering Approach", McGraw Hill, 2008.

Web Resources:

1. Software Engineering Sites: http://www.erg.abdn.ac.uk/users/brant/sre/soft-eng.html.
2. SE web - Software Engineering Education Home Page: http://tuvalu.cs.flinders.edu.au/seweb/se-ed/
3. ACM Classic Books Series: http://www.acm.org/classics/
4. Teaching Software Engineering - Lessons from MIT, by Hal Abelson and Philip Greenspun: http://philip.greenspun.com/teaching/teaching-software-engineering.
5. NASA Software Engineering Home Page:http://akao.larc.nasa.gov/dfc/swreng.html
6. Software Engineering Hotlist at Georgia Tech: http://www.cc.gatech.edu/computing/SW_Eng/hotlist.html
7. IEEE Guide to the Software Engineering Body of Knowledge: http://www.swebok.org/

16ITC19

WEB TECHNOLOGY

Instruction	3L Hours per week
Duration of End Examination	3 Hours
Semester End Examination	70 Marks
CIE	30 Marks
Credits	3

Course Objectives:

This course is introduced to

1. Acquire knowledge about design and development of web pages.
2. Develop dynamic pages using Java Servlets and JSP.
3. Know about database connectivity and how it can be used in Web-based applications.
4. Describe the state of the art of frameworks.
5. ASP.NET, to tackle challenges that are simply out of reach on many other platforms.

Course Outcomes:

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

1. Design responsive websites and validate web forms using JQuery.
2. Write a well-formed XML schemas and documents.
3. Develop dynamic web applications using Servlets and JSP.
4. Apply modern Framework techniques for web development to make applications maintainable.
5. Validate various types of controls.
6. Design and develop web applications using ASP.NET with Ajax based requests.

Prerequisites:

Java Programming (16ITC10)

UNIT-I

Introduction: Web Fundamentals, **HTML 5.0:** basic tags, Form controls, Layout Management, Graphics, Media, span and div tags.

Introduction to Cascading Style Sheets: CSS Selectors, CSS BOX Model, CSS Positioning, and CSS floating.

JQuery: Basics of JavaScript, JQuery syntax, Selectors, Events, JSON Fundamentals.

UNIT-II

Introduction to XML: The Syntax of XML, XML Document Structure, Document Type Definitions, Name Space, XML Schemas, Displaying XML Documents with CSS, XSLT Style Sheets and XML Processors.

Web Services: Web Service Architecture, structure and contents of SOAP message, structure of WSDL, Information in UDDI registry, UDDI Registry API.

UNIT-III

Java Servlets: Overview of Java Servlet API, Servlet Implementation, Servlet Configuration, Servlet Exceptions, Servlet Life cycle, Request and Response methods, Approaches to Session tracking, Request dispatching.

JSP Basics: Introduction to JSP, Directives, Scripting Elements, Standard Actions.

Database Connections: Introduction to JDBC, Database Drivers, JDBC API, connecting to my SQL, connecting to oracle, working with No SQL databases.

UNIT-IV

Struts framework: MVC Design pattern, Introduction to the modern web application framework, Architecture and flow of execution, working with actions and interceptors, Building the view with Tags and Results in detail, OGNL and type conversion, exploring the validation framework, writing the custom validator and Struts application development.

UNIT-V

ASP.NET: .Net framework, Web Form fundamentals-Html server controls, HTML control classes, Application events,ASP.net Configuration, Basic Web Control classes, State management, Building better web form - Validation, rich controls, user controls and graphics, ADO.NET Fundamentals, ASP.NET with Ajax.

Text Books:

1. Robert W.Sebesta, "Programming with World Wide Web", Eighth Edition, Pearson Education, 2008.
2. John Pollak, "JQuery - A Beginners Guide", McGraw-Hill Education, 2014.
3. Gustavo Alonso ,"Web Services: Concepts, Architectures and Applications" Springer Science & Business Media, 2004
4. Phil Hanna, "The Complete Reference JSP", First Edition, Tata McGraw-Hill, 2003
5. Donald Brown, Chad Michael Davis, Scott Stanlick, "Struts 2 in Action",Manning Publications, 2008.
6. Matthew MacDonald, "Beginning ASP.NET 4.5 in C#", Illustrated, Apress, 2012.

Suggested Reading:

1. James Webber, SavasParastatidis, Ivan Robinson," Restin Practice: HyperMedid and System Architecture", First Edition,O'REILLY,2010.
2. Deitel, Deitel, Goldberg, "Internet & World Wide Web How To Program", Third Edition, Pearson Education, 2010.
3. SubramanyamAllamraju, "Professional Java Server programming", J2EE 1.3 Edition, CeditBuest, Apress Publications

Web Resources:

1. https://msdn.microsoft.com/en-us/library/office/aa218647(v=office.11).aspx
2. https://sipb.mit.edu/iap/django/CCCDjango2010.pdf

16ITC20

THEORY OF AUTOMATA

Instruction	3L+1T Hours per week
Duration of End Examination	3 Hours
Semester End Examination	70 Marks
CIE	30 Marks
Credits	4

Course Objectives:

This course is introduced to

1. Study abstract computing models namely Finite Automata, Pushdown Automata, and Turning Machines.
2. Learn various grammars, formal languages and their relationships.
3. Learn the relation between various grammars and recognizers for different formal languages.
4. Evaluate and explain the differences between different computational models, such as Turing machines, push-down automata, finite automata, etc.
5. Familiarize with decidability and undecidability of computational problems.

Course Outcomes:

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

1. Understand formal machines, languages and design Deterministic, Nondeterministic Finite automata for acceptance of languages.
2. Build regular expressions and their equivalent finite automata for different languages.
3. Define context-free grammars for certain languages and check the ambiguity of the grammars.
4. Design pushdown automata for accepting languages.
5. DesignTuring machines for computational problems, distinguish between decidability and undecidability.

Prerequisites:

Discrete Structures and Applications (16ITC01) and Data Structures and Algorithms (16ITC02).

UNIT-IAutomata: Introduction to Finite Automata, the Central Concepts of Automata Theory: Alphabets, Strings, and Languages.

Finite Automata: An Informal Picture Of Finite Automata: The Ground Rules, the Protocol, Enabling the Automata to Ignore Actions, the Entire System as an Automaton. Deterministic Finite Automata: Definition of a DFA, Simpler Notations for DFA's, Extending the Transition Function to Strings, The Language of a DFA, Nondeterministic Finite Automata: Definition of NFA, The Extended Transition Function, The Language of an NFA, Equivalence of NFA and DFA, An Application: Text Search, Finite Automata with Epsilon-Transitions: Use of €-transitions, The formal notation for an €- NFA, €-closure, Extended Transitions and Languages for €-NFA's, Eliminating €-transitions.

UNIT-II

Regular Expression and languages: Regular Expressions: The Operators of Regular Expressions, Building Regular Expressions. Finite Automata and Regular Expression: From DFAs to Regular Expressions, Converting DFA's to Regular Expressions by Eliminating States, Converting Regular Expressions to Automata, Applications of Regular Expressions, Algebraic Laws for Regular Expressions.

Properties of Regular Languages: Proving Languages not to be Regular: The pumping lemma for Regular Languages, Applications of Pumping Lemma, Closure Properties of Regular Languages, Decision Properties of Regular Languages: Testing Emptiness of Regular Languages, Testing Membership in a Regular Language. Equivalence and Minimization of Automata: Testing Equivalence of States, Testing Equivalence of Regular Languages, Minimization of DFA's.

UNIT-III

Context Free Grammars and Languages: Context-Free Grammars: Definition of Context Free Grammars, Derivations using a Grammar, Leftmost and Rightmost Derivation, The language of a Grammar, Parse Trees: Constructing Parse Trees, The Yield of a Parse Tree, Applications of CFGs, Ambiguity in Grammars and Languages: Ambiguous Grammars, Removing Ambiguity From Grammars, Leftmost Derivations as way to Express Ambiguity, Inherent Ambiguity.

Properties of Context Free Languages: Normal Forms for Context-Free Grammars: Eliminating Useless Symbols, Computing the Generating and Reachable Symbols, Eliminating Productions, Eliminating Unit Productions, Chomsky Normal Form, Greibachnormal form, Pumping Lemma for CFL's: Statement of the Pumping Lemma, Applications of Pumping Lemma for CFL's, Closure Properties of CFL's, Decision Properties of CFL's: Testing Emptiness of CFL's, Testing Membership in a CFL's.

UNIT-IV

Pushdown Automata: Definition of pushdown automaton: The Formal Definition of PDA, Graphical Notation for PDA's, Instantaneous Description of a PDA, The Language of a PDA: Acceptance by Final State, Acceptance by Empty Stack, From Empty Stack to Final State, From Final State to Empty Stack, Equivalence of PDA's

and CFG's: From Grammars to PDA's, From PDA's to Grammars, Deterministic Pushdown Automata: Definition, Regular Languages and Deterministic PDA's, DPDA's to CFL's, DPDA's to Ambiguous Grammars.

UNIT-V

Introduction to Turing Machines: Problems that Computer Cannot Solve: The Turing Machine: Notation for the TM, Instantaneous Descriptions for TM's, Transitions Diagrams, The Language of a TM, Turing Machines and Halting, Programming Techniques for Turing Machines: Storage in the State, Multiple Tracks, Subroutines, Extensions to the Basic Turing Machine: Multitape Turing Machine, Equivalence of One-Tape and Multi-Tape TM's, Nondeterministic Turing Machines, Restricted Turing Machines: TM's with Sem infinite Tapes, Multistack Machines, Counter Machines. Turing Machine and Computers: Simulating a Computer by a TM.

Undecidability: A Language That Is Not Recursively Enumerable: Enumerating the Binary Strings, Codes for Turing Machines, The Diagonalization Language, An Undecidable problem that is RE: Recursive Languages, Compliments of Recursive and RE languages, The Universal Languages, Undecidablity of the Universal Language, Undecidable problems about Turing Machines: Reductions, TM's That Accept The Empty Language, Rice's Theorem and Properties of RE languages, Post's Correspondence Problem: Definition of PCP, The Modified PCP, Other Undecidable Problems.

Text Book:

1. John E. Hopcroft, Rajeev Motwani, Jeffery D Ullman, "Introduction to Automata Theory Languages and Computation", Third edition, Pearson Education, 2007.

Suggested Reading:

1. John C Martin. "Introduction to Language and Theory of Computation", 3rdedition, TMH, 2003.
2. Daniel Cohen, "Introduction to Computer Theory", 2ndedition, Wiley Publications, 2007.
3. Mishra K., Chandrasekaran N., "Theory of Computer Science (Automata, Languages and Computation)", 3rd edition, Prentice Hall of India 2008.
4. ShyamalendraKandar, "Introduction to Automata Theory, Formal Languages and Computation", Pearson, 2013.
5. Kamala Krithivasan, Rama R. "Introduction to Automata Theory, and Computation", Pearson 2009.

Web Resources:

1. http://nptel.ac.in/courses/106106049/
2. http://online.stanford.edu/course/automata-theory
3. https://www.tutorialspoint.com/automata_theory/

16ITE01

PYTHON PROGRAMMING
(Elective - I)

Instruction	3L Hours per week
Duration of End Examination	3 Hours
Semester End Examination	70 Marks
CIE	30 Marks
Credits	3

Course Objectives:

This course is introduced to

1. Familiarize the fundamentals of Python programming
2. Learn how to use lists, tuples, and dictionaries in Python programs
3. Learn how to read and write files in Python
4. Impart usage of exception handling in Python
5. Familiarize data visualization

Course Outcomes:

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

1. Understand basic data structures of python
2. Understand the concepts of file I/O
3. Understand exception handling in Python.
4. Develop proficiency in creating GUI based applications
5. Plot data using appropriate Python visualization libraries
6. Develop simple Python applications.

Prerequisites:

Programming and Problem Solving (16CSC01), Programming Laboratory (16CSC02)

UNIT-I

Introduction to Python Programming: Using Python, The IDLE Programming Environment, Input and Output Processing, Displaying Output with the Print Function, Comments, Variables, Reading Input from the Keyboard, Performing Calculations, More About Data Output: New line, Item Separator, Escape Characters, Formatting parameters.

Decision Structures and Boolean Logic: if, if-else, if-elif-else Statements, Nested Decision Structures, Comparing Strings, Logical Operators, Boolean Variables.

Repetition Structures: Introduction, while loop, for loop, Sentinels, Input Validation Loops, Nested Loops.

UNIT-II

Functions: Introduction, Defining and Calling a Function, Designing a Program to Use Functions, Local Variables, Passing Arguments to Functions, Global Variables and Global Constants, Value-Returning Functions Generating Random Numbers, Writing Our Own Value-Returning Functions, The math Module, Random Module, Time Module and Storing Functions in Modules.

UNIT-III

Lists and Tuples: Sequences, Introduction to Lists, List slicing, Finding Items in Lists with the in Operator, List Methods and Useful Built-in Functions, Copying Lists, Processing Lists, Two-Dimensional Lists, Tuples.

Strings: Basic String Operations, String Slicing, Testing, Searching, and Manipulating Strings.

Dictionaries and Sets: Dictionaries, Sets, Serializing Objects.

Recursion: Introduction, Problem Solving with Recursion, Examples of Recursive Algorithms.

UNIT-IV

Python File Input-Output: Opening and closing file, various types of file modes, reading and writing to files, manipulating directories

Exception Handling: What is exception, various keywords to handle exception such try, catch, except, else, finally, raise.

Regular Expressions: Concept of regular expression, various types of regular expressions, using match function.

UNIT-V

GUI Programming: Graphical User Interfaces, Using the tkinter Module, Display text with Label Widgets, Organizing Widgets with Frames, Button Widgets and Info Dialog Boxes, Getting Input with Entry Widget, Using Labels as Output Fields, Radio Buttons, Check Buttons.

Introduction to plotting in Python – Basic Plots- Line and Scatter Plot, Histograms and plotting data contained in files.

Text Books:

1. Tony Gaddis, "Starting Out With Python", 3rd edition, Pearson, 2015.
2. Charles Dierbach, "Introduction to Computer Science using Python", Wiley, 2013.

Suggested Reading:

1. Kenneth A. Lambert, "Fundamentals of Python", Delmar Cengage Learning, 2013.
2. James Payne, "Beginning Python using Python 2.6 and Python 3", wrox programmer to programmer, 2010.

3. Paul Gries, "Practical Programming: An Introduction to Computer Science using Python",3rd edition, 2016.
4. Clinton W. Brownley, "Foundations for Analytics with Python", 1st edition, O'Rielly Media, 2016.

Web Resources:

1. https://www.python.org/
2. https://www.coursera.org/learn/python
3. https://learnpythonthehardway.org/book/
4. https://www.coursera.org/specializations/python

16ITE02

UNIX AND SHELL PROGRAMMING
(Elective - I)

Instruction	3L Hours per week
Duration of End Examination	3 Hours
Semester End Examination	70 Marks
CIE	30 Marks
Credits	3

Course Objectives:

This course is introduced to

1. Familiarize students with the UNIX environment and basic UNIX utilities
2. Learn File systems and File structures.
3. Impart skills required to write shell scripts.
4. Develop skills required to formulate regular expressions.
5. Familiarize students with the routine system administrative features and tools.

Course Outcomes:

Upon successful completion of this course, the students should be able to

1. Understand the UNIX architecture, basics of vi editor and UNIX utilities.
2. Implement various File processing commands, change file permissions and directory permissions.
3. Create and manage processes using the knowledge of process attributes process creation and process control mechanisms.
4. Construct simple and complex shell scripts to automate jobs and processes in UNIX environment.
5. Locate and replace patterns at specific locations using regular expressions
6. Demonstrate administrator privileges, super user basic commands to add, modify and delete users.

Prerequisites: Programming and Problem Solving (16CSC01), Programming Laboratory (16CSC02).

UNIT-I

Introduction to Unix: The UNIX Operating System, The UNIX Architecture, Features of UNIX, Internal and External Commands, Command Structure,

General-Purpose Utilities:cal, date, echo, printf, bc, script, mailx, passwd, who, uname, tty, sty,

The vi editor: vi Basics, Input Mode, Saving Text and Quitting, Navigation, Editing Text, Undoing Last Editing Instructions, Repeating the Last Command, Searching for a Pattern, Substitution.

UNIT-II

Handling Files: The File System, Parent Child Relationship, The HOME variable, pwd,cd, mkdir, rmdir, Absolute Pathnames, Relative Pathnames, The UNIX File System cat, cp, rm, mv, more, file, ls, wc, cmp, comm, diff,

Compressing and Archiving files: gzip and gunzip- Compressing and Decompressing files, tar- The Archival program, zip and unzip- Compressing and Archiving together.

File Attributes: ls options -l, -d,-lh,-la, File Ownership, File Permissions, chmod-Changing File permissions, Directory Permissions, Changing File ownership.

UNIT-III

The Shell: The Shells's interpretive Cycle, Shell Offerings, Pattern Matching, Escaping and quoting, Redirection, /dev/null and /dev/tty, Pipes, tee- Creating a tee, Command Substitution, Shell Variables.

The Process: Process Basics, ps- Process Status, System Processes (-e or –a), Mechanism of Process creation, Internal and External Commands, Process states and Zombies, Running jobs in Background, nice-Job Execution with low priority, Killing Processes with signals, Job Control, at and batch-Execute later, cron- Running jobs periodically, time-Timing Processes.

UNIT-IV

Simple Filters: pr-Paginating Files, head- Displaying the beginning of a File, tail-Displaying the end of a File, cut- Slitting a File vertically, paste-Pasting Files, sort-Ordering a File, uniq- Locate Repeated and Non-repeated Lines, tr- Translating Characters.

Filters using Regular Expressions: grep, Basic Regular Expressions, Extended Regular expressions, egrep, sed, Line Addressing, Using multiple instructions, Context Addressing, Writing Selected lines to a File, Text Editing, Substitution.

UNIT-V

Shell Programming: Shell scripts, read, Using Command Line Arguments, exit, The logical operators && and ||, Conditional execution- if, Using test and [ ] to evaluate expressions, case, expr, while, for, set and shift, trap, Debugging shell scripts with set–x.

System Administration: root, The administrator's privileges, Maintaining Security, User Management, Startup and Shutdown, Managing Disk Space, Device Files.

Text Book:

1. Sumitabha Das, "Unix Concepts and Applications", 4th Edition, TMH, 2006.

Suggested Reading:

1. Behrouz A. Forouzan, Richard F. Gilbery, "Unix and Shell Programming", 1st Edition, Cengage Learning India, 2003.

2. Graham Glass, King Ables, "Unix for programmers and users", 3rd Edition, Pearson Education, 2009.
3. YashwanthKanitkar, "Unix Shell programming", 1st Edition, BPB Publishers, 2010.

Web Resources:

1. http://openclassroom.stanford.edu/MainFolder/CoursePage.php?course=PracticalUnix
2. https://www.shellscript.sh/
3. www.bash.academy/
4. http://linuxcommand.org/

16ITE03

SCRIPTING LANGUAGES
(Elective - I)

Instruction	3L Hours per week
Duration of End Examination	3 Hours
Semester End Examination	70 Marks
CIE	30 Marks
Credits	3

Course Objectives:

This course is introduced to

1. Write scripts to extract meaningful summaries from partially structured text.
2. Prepare students to use Python to perform common scripting tasks.
3. Allow students to use scikit-image library to learn image processing algorithms.
4. Familiarize students with PHP for making dynamic and interactive web pages.
5. Improve VB Scripting Skills for writing desktop, web applications and automation of tasks.

Course Outcomes:

Upon successful completion of this course, the students should be able to

1. Use Perl language features in web application development
2. Master the fundamentals of writing Python scripts
3. Implement algorithms and techniques involved in Digital Image Processing using scikit-image package
4. Gain the PHP programming skills needed to successfully build interactive, data-driven websites.
5. Use Ajax technology to load new content without leaving the current page, creating a better, faster experience for webpages
6. Develop web, desktop and various automation tasks using Visual Basic Scripting (VB Script)

Prerequisites:

Programming and Problem Solving (16ITC01), Data structures and algorithms (16ITC02)

UNIT-I

PERL- Names and Values, Variables, Scalars, Arrays and its operations, Hashes, Regular expressions, string manipulation, File management, Command line arguments, sub routines, Packages, Modules.

UNIT-II

Introduction to Python: Variables, Lists and Tuples, Introducing Functions , If statements, While Loops and Input, Basic Terminal Apps, Dictionaries, More Functions, Classes and OOPs, Exceptions.

UNIT-III

Simple Graphics and Image Processing using Python: "turtle" module; simple 2d drawing - colors, shapes; digital images, image file formats, image processing Simple image manipulations with 'image' module (convert to between, grey scale, blur, etc). Graphical user interfaces; event-driven programming paradigm; tkinter module, creating simple GUI; buttons, labels, entry fields, dialogs; widget attributes - sizes, fonts, colors layouts, nested frames.

UNIT-IV

Programming with PHP: PHP Basics, String Manipulation and regular expressions, Form handling, Adding dynamic Content, Managing Web sessions, Handling Date & Time in PHP, Sending email with PHP, Object Oriented Programming and PHP7, Exception handling, Accessing Databases using PHP, AJAX with PHP.

UNIT-V

VBScript: Introduction to VBScript, Declaring and Using Variables, Operators, Operator Precedence and Constants, Using Conditional Statements, Loops in VBScript, Using Procedures and Functions, Arrays, Date Functions, Working with Strings and Cookies, Working with Events, Working with Excel Objects, Working with Connection Objects, Working with Files, Error Handling

Text Books:

1. Randal L. Schwartz, Tom Phoenix, brianfoy, "Learning Perl", 5th Edition, O'Reilly Media, 2008.
2. Kenneth A. Lambert , "Fundamentals of Python First Programs", Cengage Learning, 2012.
3. Luke Welling, Laura Thomson, "PHP and MySQL Web Development", Pearson Education, 2017.
4. Kogent Solutions Inc, "Ajax Black Book", Dreamtech press, 2008.
5. Adrian Kingsley-Hughes, Kathie Kingsley-Hughes, Daniel Read, "VBScript – Programmers Reference", 3rd Edition, wiley publications, 2007.

Suggested Reading:

1. John ericsolem, "Programming Computer Vision with Python", First edition, O'Reilly Media, 2012
2. Thomas A Powel, "The Complete Reference: AJAX", 1st Edition, Tata McGraw Hill, 2008.

Web Resources:

1. https://docs.python.org/3/tutorial/
2. https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-0001-introduction-to-computer-science-and-programming-in-python-fall-2016/
3. https://learn.perl.org/

16ITC21

OPERATING SYSTEMS AND WEB TECHNOLOGY LAB

Instruction	3L Hours per week
Duration of End Examination	3 Hours
Semester End Examination	50 Marks
CIE	25 Marks
Credits	2

Course Objectives:

This course is introduced to

1. Familiarize with various system calls of LINUX
2. Learn processes synchronization and scheduling algorithms
3. Design and develop web pages using html5, CSS positioning, servlets and JDBC.
4. Learn and write a well-formed XML schemas and documents.
5. Learn MVC based web application development using Struts2 and ASP.NET.

Course Outcomes:

Upon successful completion of this course, the students should be able to

1. Create multiple processes and replace a process image using different system calls.
2. Understand Inter-process communication using shared memory, message passing and pipes.
3. Analyze and evaluate different algorithms for CPU scheduling.
4. Design various web based applications using HTML5,Jquery and CSS.
5. Use JDBC, JSP and Struts 2 framework, to build modem web applications.
6. Design web site using ASP.NET with Ajax based requests.

Prerequisites:

Programming Laboratory (16CSC02), Data Structures and Algorithms Lab (16ITC05), Java Programming Lab (16ITC13).

List of Programs

1. Demonstrate the following system calls:
   a) fork b) execvp c) stat d) setenv&getenv
2. Implement Echo Server using
   a) Pipes b) Shared memory c)Message queues

3. Simulate the following CPU Scheduling Algorithm:
   a) FCFS b) SJF c) Round Robin
4. Implement Producer-Consumer Problem using
   a) Message passing b) Semaphores
5. Develop an e-commerce web site having the following specifications
   a) Use css for styling all the web controls.
   b) Use jquery for all form validations.
   c) All form submissions should be with AJAX only.
   d) Use menus in appropriate places.
6. Write a DTD and Schema for a library management system and give an XML example for each.
7. Build a java based dynamic working e-commerce website mentioned in question no.5 with database connections.
8. Develop a struts2 framework based "registration and login" application making use of validator framework.
9. Design and develop a simple web based application for "online quiz management" using ASP.NET.
10. Write an application to demonstrate data management using ADO.NET.

Text Books:

1. W. Richard Stevens, "Unix Network Programming", Volume 2, 2nd edition, Pearson Education, 2015.
2. Robert W.Sebesta, "Programming with World Wide Web", Eighth Edition, Pearson Education, 2008.
3. John Pollak, "JQuery - A Beginners Guide", McGraw-Hill Education, 2014.
4. Phil Hanna, "The Complete Reference JSP", First Edition, Tata McGraw-Hill, 2003.
5. Matthew MacDonald, "Beginning ASP.NET 4.5 in C#", Illustrated, Apress, 2012.

Suggested Reading:

1. Silberschatz, Galvin, and Gagne,"Operating System Concepts", 8thEdition,Wiley Publication.
2. Andrew S. Tanenbaum, "Modern Operating Systems", 3rd Edition, GOAL Series.
3. James Webber, SavasParastatidis, Ivan Robinson,"Rest in Practice: HyperMedid and System Architecture", First Edition,O'REILLY,2010.

4. Deitel, Deitel, Goldberg, “Internet & World Wide Web How To Program”, Third Edition, Pearson Education, 2010.
5. SubramanyamAllamraju, “Professional Java Server programming”, J2EE 1.3 Edition, CeditBuest, Apress Publications.

Web Resources:

1. http://www.tutorialspoint.com/unix/
2. https://msdn.microsoft.com/en-us/library/office/aa218647(v=office.11).aspx

16ITC22

DATABASE SYSTEMS LAB

Instruction	3L Hours per week
Duration of End Examination	3 Hours
Semester End Examination	50 Marks
CIE	25 Marks
Credits	2

Course objectives:

This course is introduced to

1. Present the concepts and techniques relating to query processing.
2. Design and develop database for an application.
3. Learn the basic commands, SQL functions and the significance of triggers.
4. Learn how to manipulate a database using SQL.
5. Familiarize with the various methods of database security.

Course outcomes:

Upon successful completion of this course, the students should be able to:

1. Design and implement database schemas by enforcing integrity constraints for a given problem domain.
2. Use SQL for database administration(to create tables, indexes, and views) and data manipulation.
3. Write efficient data retrieval queriesusing relational set operators and advanced SQL Join operators.
4. Do PL/SQL programming and define various triggers and cursors for the databases.
5. Create Security features and facilities for the database applications.
6. Design, create, and test data entry forms and detailed reports that require access to data in multiple tables.

Prerequisites:

Programming and Problem Solving (16CSC01)

List of Programs

1. Creation of database (Exercising commands like DDL and DML)
   (Note: use constraints while creating tables).
2. Exercising simple to complex queries.

a. Queries (along with sub Queries) using ANY, ALL, IN, EXISTS, NOTEXISTS, UNION, INTERSECT Constraints.
b. Queries using Aggregate functions (COUNT, SUM, AVG, MAX and MIN), GROUP BY, HAVING clause and Creation and dropping of Views.
c. Exercising all types of Joins.

3. Demonstration of PL/SQL Blocks and Cursors.
4. Demonstration of Procedures and Functions.
5. Usage of Triggers (Programs using BEFORE and AFTER Triggers, Row and Statement level Triggers and INSTEAD OF Triggers).
5. Demonstrate Exception Handling by PL/SQL procedures for data validation.
6. Creating Password and Security features for applications.
7. Usage of File locking table locking, facilities in applications.
8. Creation of Forms and Generation of SQL reports.
9. Creation of full-fledged database application spreading over to 3 sessions.

Note:-The creation of sample database for the purpose of the experiments is to be pre-decided by the instructor.

Text Book:

1. Rick F Vander Lans, "Introduction to SQL", Fourth edition, Pearson Education, 2007.

Suggested Reading:

1. Benjamin Rosenzweig, Elena Silvestrova, "Oracle PL/SQL by Example", Fifth Edition, Pearson Education, 2015.
2. Albert Lulushi, "Oracle Forms Developer's Handbook", Pearson Education, 2006.

Web Resources:

1. http://www.oracle-dba-online.com/sql/oracle_sql_tutorial.htm.
2. https://www.javatpoint.com/sql-tutorial
3. https://www.tutorialspoint.com/sql/
4. http://www.tutorialspoint.com/plsql/
5. https://www.javatpoint.com/pl-sql-tutorial

16ITC23

MINI PROJECT - III

Instruction	2 Hours per week
Duration of End Examination	-
Semester End Examination	-
CIE	50 Marks
Credits	1

Course Objectives:

1. To enable students learn by doing.
2. To develop capability to analyse and solve real world problems.
3. To develop innovative ideas among the students.

Course Outcomes:

Students should be able to do the following:

1. To provide innovative solutions.
2. To work in a team.
3. To manage time and resources in the best possible manner.

The Students are required to implement one of the projects from project exercise given in the suggested readings of the theory subjects of the current semester / as suggested by the respective course faculty of that semester. During the implementation of the project, Personnel Software Process (PSP) has to be followed. Report of the project work has to be submitted for evaluation.

CHAITANYA BHARATHI INSTITUTE OF TECHNOLOGY(A)
Choice Based Credit System (with effect from 2018-19)
B.E (Information Technology)

SEMESTER – VI

S.No	Course Code	Title of the Course	Scheme of Instruction		Scheme of Examination			Credits
			Hours per week		Duration of SEE in Hours	Maximum Marks		
			L/T	P/D		CIE	SEE	
THEORY								
1	16ITC24	Computer Networks & Socket Programming	3/1	-	3	30	70	4
2	16ITC25	Data Warehousing and Data Mining	3	-	3	30	70	3
3	16ITC26	Artificial Intelligence	3	-	3	30	70	3
4	16ITC27	Principles of Compiler Design	3/1	-	3	30	70	4
5		Elective – II	3	-	3	30	70	3
6		Elective – III	3	-	3	30	70	3
PRACTICALS								
7	16ITC28	Network Programming Lab	-	3	3	25	50	2
8	16ITC29	Data Mining Lab	-	3	3	25	50	2
9	16ITC30	Mini Project-IV	-	2	-	50	-	1
TOTAL			**20**	**8**	**-**	**280**	**520**	**25**

L: Lecture T: Tutorial D: Drawing P: Practical
CIE - Continuous Internal Evaluation SEE - Semester End Examination

Elective-II

S.No.	Subject Code	Subject Name
1.	16ITE04	Principles of Computer Graphics
2.	16ITE05	File Structures
3.	16ITE06	Object Oriented System Development using UML

Elective-III

S.No.	Subject Code	Subject Name
1.	16ITE07	Digital Image Processing
2.	16ITE08	Information Retrieval Systems
3.	16ITE09	E-Commerce

16ITC24

COMPUTER NETWORKS AND SOCKET PROGRAMMING

Instruction	3L+1T Hours per week
Duration of End Examination	3 Hours
Semester End Examination	70 Marks
CIE	30 Marks
Credits	4

Course Objectives:

This course is introduced to

1. Familiarize students with basics of Socket based Client/Server programming.
2. Provide state-of-the-art knowledge on Network Layer issues including Routing, Addressing, Congestion Control and Quality of Service.
3. Give an overview of how Networks differ and how they can be interconnected.
4. Introduce IP based transport protocols TCP and UDP.
5. Give an insight into the working principles of popular Internet Applications including Email and Domain Name System.
6. Provide a solid understanding of main issues related to network security and the relevant cryptographic techniques.

Course Outcomes:

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

1. Enumerate functions of each layer in the OSI and TCP/IP reference models and build Client/Server applications using the understanding of Socket System calls.
2. Solve problems related to Addressing, Routing and Congestion in computer networks.
3. Understand Internetwork Routing issues and Interoperability among heterogeneous networks.
4. Analyze the functions and performance of Internet Transport Protocols TCP and UDP.
5. Understand the operating principles of Domain Name System and Electronic Mail.
6. Comprehend various network security threats and cryptographic algorithms.

Prerequisites:

Data Communications (16ITC09), Programming and Problem Solving (16CSC01).

UNIT-I

Introduction: Uses of Computer Networks, ISO/OSI and TCP/IP Reference Models, Comparison of the OSI and TCP/IP Reference Models.

Socket programming: Socket address, Elementary socket system calls, Advanced socket system calls, Reserved ports, Socket options, Asynchronous I/O, Out-of-Band data, Internet Super Server, Daemon Processes.

UNIT-II

Network Layer Design Issues: Store and Forward Packet switching, Services, Implementation of Connectionless Service and Connection-Oriented Service, Comparison of Virtual circuits and Datagram subnets.

Routing Algorithms: The Optimality principle, Shortest path routing, Flooding, Distance vector Routing, Link state Routing, Hierarchical Routing, Broadcast and Multicast routings,

Congestion control algorithms: Approaches, Traffic-Aware Routing, Admission Control, Traffic Throttling, Load Shedding,

Quality of Service: Application Requirements, Traffic shaping Packet Scheduling, Integrated and Differentiated Services.

UNIT-III

Internetworking: How networks differ, How networks can be Connected, Tunneling, Internetwork routing, Packet Fragmentation,

The Network Layer in the Internet: The IPv4 protocol, IP addresses, Subnets, Classless Inter Domain Routing, Classful and Special Addressing, Network Address Translation, IP version 6, Label Switching and MPLS, OSPF, BGP.

UNIT-IV

Transport Layer: Transport service primitives, Addressing, Connection Establishment, Connection Release, Error Control and Flow control, Multiplexing and Crash recovery.

Internet Transport Protocols (TCP and UDP): Introduction to UDP, Remote Procedure Call (RPC), Real-Time Transport Protocols, The TCP service model, The TCP protocol, The TCP Segment Header, TCP Connection Establishment, TCP Connection Release, TCP Connection Management Modeling, TCP Sliding Window, TCP Timer management, TCP Congestion Control, Performance issues.

UNIT-V

Application Layer: The Domain Name System- DNS Name Space, Domain Resource Records, Name Servers, Electronic Mail-Architecture and Services, The User Agent, Message Transfer, SMTP and Extensions, Final Delivery,

Network Security: Introduction to Cryptography, Substitution Ciphers, Transposition Ciphers, Symmetric Key Algorithms-The Data Encryption Standard (DES), Triple DES, Public Key Algorithm:RSA Algorithm, Digital Signatures:

Symmetric-Key Signatures, Public-Key Signatures, Message Digests, Authentication Protocols.

Text Books:

1. Andrew S. Tanenbaum, David J. Wetherall, "Computer Networks", 5th Edition, Pearson Education, 2014.
2. W. Richard Stevens, Unix Network Programming, Prentice Hall/Pearson Education, 2009.

Suggested Reading:

1. Chwan-Hwa (John) Wu, J. David Irwin, Introduction to Computer Networks and Cyber Security, CRC Press, 2013.
2. James F. Kurose and Keith W. Ross, Computer Networking: A Top-Down Approach Featuring the Internet, 5th Edition, Addison-Wesley, 2012.

Web Resources:

1. http://www.nptelvideos.in/2012/11/computer-networks.html
2. beej.us/guide/bgnet/output/print/bgnet_A4.pdf

16ITC25

DATA WAREHOUSING AND DATA MINING

Instruction	3LHours per week
Duration of End Examination	3 Hours
Semester End Examination	70 Marks
CIE	30 Marks
Credits	3

Course Objectives:

This course is introduced to

1. Familiarise the concepts of Data Warehouse and Data Mining techniques.
2. Examine the types of the data to be mined and apply preprocessing methods on raw data.
3. Present different frequent pattern discovery methods.
4. Describe various classification and clustering techniques.
5. Minecomplex data types.

Course Outcomes:

Upon successful completion of this course, the students should be able to

1. Understand requirements of data warehousing and data mining to the decision support level of organizations.
2. Apply Pre-Processing techniques on various data formats to make it suitable for data mining algorithms.
3. Generate Association rules for the data.
4. Build models for Classification, prediction, and clustering.
5. Evaluate the performance of various data mining algorithms.
6. Understand mining of complex data.

Prerequisites:

Database Systems (16ITC17), Database Lab (IT 317).

UNIT-I

Introduction: What is Data mining? What kinds of data can be mined? What kinds of pattern can be mined? Major issues in data mining.

Getting to know your data: Data Objects and Attribute Types, Basic Statistical Descriptions of Data, Measuring Data Similarity and Dissimilarity.

Data Preprocessing: An Overview, Data Cleaning, Data Integration, Data Reduction, Data Transformation and Data Discretization.

UNIT-II

Data Warehousing and Online Analytical Processing Data Warehouse: Basic Concepts, Data Warehouse Modeling: Data Cube and OLAP, Data Warehouse Design

and Usage: A Business Analysis Framework for Data Warehouse Design, Data Warehouse Design Process, Data Warehouse Usage for Information Processing, Data Warehouse Implementation.
Mining Frequent Patterns, Associations and correlations: Basic Concepts, Frequent Item Set Mining Methods, Interesting patterns, Pattern Evaluation Methods.
Advanced Pattern Mining: Pattern Mining in Multilevel and multidimensional space.

UNIT-III
Classification: Basic Concepts**,** Decision Tree Induction, Bayes Classification Methods, Rule-Based Classification, Model Evaluation and Selection, Techniques to Improve Classification Accuracy: Introducing Ensemble Methods, Bagging, Boosting, Random Forests, Improving Classification Accuracy of Class-Imbalanced Data. **Classification: Advanced Methods** Bayesian Belief Networks, Classification by Back propagation, Support Vector Machines, Lazy Learners (or Learning from Your Neighbors), Other Classification Methods.

UNIT-IV
Cluster Analysis: Basic Concepts and Methods: Cluster Analysis, Partitioning Methods, Hierarchical Methods: Agglomerative versus Divisive Hierarchical Clustering, Distance Measures in Algorithmic Methods, DBSCAN, Evaluation of Clustering.

UNIT-V
Outlier Detection: Outliers and Outlier Analysis, Outlier Detection Methods, Statistical Approaches, Proximity-Based Approaches
Data Mining Trends and Research Frontiers: Mining Complex Data Types: Mining Sequence Data: Mining Other Kinds of Data, Data Mining Applications, Data Mining and Society, Data Mining Trends.

Text Book:

1. Han J, Kamber M, Jian P "Data Mining: Concepts and Techniques", Third Edition, Elsevier, 2012.

Suggested Reading:

1. Pang-Ning Tan, Michael Steinback, Vipin Kumar, "Introduction to Data Mining", Pearson Education, 2008.
2. M. Humphires, M.Hawkins, M.Dy,"Data Warehousing: Architecture and Implementation", Pearson Education, 2009.
3. Anahory, Murray, "Data Warehousing in the Real World", Pearson Education, 2008.
4. Kargupta, Joshi, etc., "Data Mining: Next Generation Challenges and Future Directions", Prentice Hall of India Pvt. Ltd, 2007.

Web Resources:

1. https://www.kdnuggets.com/
2. http://archive.ics.uci.edu/ml/index.php

16ITC26

ARTIFICIAL INTELLIGENCE

Instruction	3L Hours per week
Duration of End Examination	3 Hours
Semester End Examination	70 Marks
CIE	30 Marks
Credits	3

Course Objectives:

This course is introduced to

1. Learn problem solving techniques.
2. Familiarize with knowledge representation and logical reasoning techniques used in Artificial Intelligence.
3. Learn probabilistic reasoning models on uncertain data.
4. Design machine learning and neural network systems.
5. Learnsyntax and semantics of the natural language.

Course Outcomes:

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

1. Solve problems using Exhaustive and Heuristic Search Techniques.
2. Apply inference methods in propositional logic to prove statements.
3. Apply probabilistic reasoning models on uncertain data.
4. Apply classification and clustering techniques on data sets.
5. Understand the working of neural networks to store and process information
6. Understand syntax and semantics of the language and knowledge representations.

Prerequisites:

Discrete Structures and Applications (16ITC01), Fundamentals of Data Science (16ITC12).

UNIT-I

Introduction – The Foundations of AI, History of AI.

Intelligent agents – Agents and Environments, Good Behavior: The Concept of Rationality, The Nature of Environments, The Structure of Agents.

Solving problems by searching – Problem Solving Agents, Example Problems, Searching for Solutions, Uninformed Search Strategies, Informed Search Strategies, Heuristic Functions.

Adversarial search – Games, Optimal decisions in games, Alpha-Beta Pruning.
Constraint Satisfaction Problems- Defining constraint satisfaction Problems.

UNIT-II

Logic Concepts and Logic Programming: Introduction, Propositional Calculus, Propositional Logic, Natural Deduction System, Axiomatic System, Semantic Tableau System in Propositional Logic, Resolution Refutation in Propositional Logic, Predicate Logic, Logic Programming.

Knowledge Representation: Introduction, Approaches to Knowledge Representation, Knowledge Representation using Semantic Network, Extended Semantic Networks for KR, Knowledge Representation using Frames.

UNIT-III

Quantifying Uncertainty- Acting under Uncertainty, Basic Probability Notation, Inference Using Full Joint Distributions, Independence, Bayes' Rule and its Use.
Probabilistic Reasoning - Representing Knowledge in an Uncertain Domain, The Semantics of Bayesian Networks, Efficient Representation of Conditional Distributions, Exact Inference in Bayesian Networks.

Probabilistic Reasoning over Time-Time and Uncertainty, Inference in Temporal Models, Hidden Markov Models, Kalman Filters.

UNIT-IV

Learning from Examples- Forms of Learning, Supervised Learning, Learning Decision Trees, Evaluating and Choosing the Best Hypothesis, The Theory of Learning, Regression and Classification with Linear Models, Artificial Neural Networks, Nonparametric Models, Support Vector Machines.
Learning Probabilistic Models- Statistical Learning, Learning with Complete Data.
Learning with Hidden Variables: The EM Algorithm

UNIT-V

Natural Language Processing-Language Models, Text Classification, Information Retrieval, Information Extraction.
Natural Language for Communication-Phrase Structure Grammars, Syntactic Analysis, Augmented Grammars and Semantic Interpretation.

Text Books:

1. Russell, Norvig, "Artificial intelligence - A Modern Approach", Pearson Education, Third Edition, 2015.
2. Saroj Kaushik, "Artificial Intelligence", Cengage Learning, 2011.

Suggested Reading:

1. Nilsson, N., “Artificial Intelligence: A New Synthesis”, San Francisco, Morgan Kaufmann, 1998.
2. Rich, Knight, Nair: “Artificial intelligence”, Tata McGraw Hill, Third Edition, 2009.
3. Tom M. Mitchell, “Machine Learning”, McGraw Hill, 1997.
4. Kulkarni, Parag, Joshi, Prachi ,”Artificial Intelligence : Building Intelligent Systems”, PHI, 2015.
5. Peter Jackson, “Introduction to Expert Systems”, Third Edition, Pearson Addison Wesley, 1998.

Web Resources:

1. http://www.nptel.ac.in/courses/106105077/
2. https://www.coursera.org/specializations/machine-learning

16ITC27

PRINCIPLES OF COMPILER DESIGN

Instruction	3L+1T Hours per week
Duration of End Examination	3 Hours
Semester End Examination	70 Marks
CIE	30 Marks
Credits	4

Course Objectives:

This course is introduced to

1. Learn various phases of Compiler Design.
2. Design scanner and Parsers.
3. Develop Intermediate code and generate code for target machine.
4. Familiarize with machine dependent and machine independent optimization techniques.
5. Present the role of a symbol table and error recovery strategies.

Course Outcomes:

Upon successful completion of this course, the students should be able to

1. Understand various phases in the design of compiler.
2. Generate a lexical analyser.
3. Design top-down and bottom-up parsers.
4. Develop Syntax Directed Translation scheme and Generate Intermediate code for a language.
5. Develop algorithms to generate code for a target machine.
6. Understand Data flow Analysis and Apply the optimization techniques.

Prerequisites:

Programming and Problem Solving (16CSC01), Data Structures and Algorithms (16ITC02),
Theory of Automata (16ITC20).

UNIT-I

Introduction: Programs related to compilers, Translation process, Major data structures, Other issues in compiler structure, Boot strapping and porting.

Lexical analysis: The role of Lexical Analyzer, Input Buffering, Specification of Tokens, Recognition of Tokens, The Lexical-Analyzer Generator Lex.

UNIT-II

Syntax Analysis: Introduction, Context-Free Grammars, Writing a Grammar, Top-Down parsing, Bottom-Up parsing, Introduction to LR Parsing, More powerful LR parsers, Using Ambiguous Grammars, Parser Generators.

UNIT-III

Syntax Directed Translation: Syntax Directed Definitions, Evaluation Orders for SDDs, Applications of Syntax Directed Translation.

Intermediate code generation: Variants of Syntax Trees, Three-Address Code, Types and Declarations, Translation of Expressions, Type Checking, Control Flow.

UNIT-IV

Runtime Environments: Storage Organization, Stack Allocation of Space, Access to Non local Data on the Stack, Heap Management, Introduction to Garbage Collection.

Code Generation : Issues in the Design of a Code Generator, The Target Language, Addresses in the Target Code, Basic Blocks and Flow Graphs, Optimization of Basic Blocks, Peephole Optimization, Register Allocation and Assignment.

UNIT-V

Machine Independent Optimizations: The Principal Sources of Optimizations, Introduction to data flow analysis, Foundation of data flow analysis.

Text Books:

1. Alfred V Aho, Monica S Lam, Ravi Sethi, Jeffrey D Ullman ,"Compilers: Principles, Techniques &Tools", Pearson Education, Second Edition, 2014.
2. Kenneth C Louden, "Compiler Construction: Principles and Practice", Cengage Learning.

Suggested Reading:

1. Keith D Cooper & Linda Torczon, "Engineering a Compiler", Morgan Kafman, Second Edition.
2. Dick Grune, Kees van Reeuwijk, Henri E. Bal , Ceriel J.H. Jacobs, Koen Langendoen ," Modern Compiler Design", Springer, Second Edition.

Web Resources:

1. http://nptel.ac.in/courses/106108113

16ITE04

PRINCIPLES OF COMPUTER GRAPHICS
(Elective-II)

Instruction	3LHours per week
Duration of End Examination	3 Hours
Semester End Examination	70 Marks
CIE	30 Marks
Credits	3

Course objectives:

This course is introduced to

1. Explain the core concepts of computer graphics.
2. Displaying two dimensional output primitives for raster graphics system.
3. Acquire knowledge about transformation techniques in 2D and 3D.
4. To learn various algorithms on clipping techniques.
5. To acquire knowledge about curve generation and animations.

Course outcomes:

Students who complete this course should be able to

1. Understand the core concepts of computer graphics.
2. Understand the techniques for performing 2D and 3D transformations.
3. Describe various techniques for clipping.
4. Demonstrate problem solving skills with application to computer graphics.
5. Understand graphics techniques for curve generation.
6. Explain fundamentals of shading and animation techniques.

Prerequisites: Engineering Mathematics-I (16MTC01)

UNIT-I

Computer Graphics: Introduction, Application areas, Overview of graphics systems: Video-display devices, Raster-scan systems, Random scan systems, Graphics monitors and Work stations and input devices, Graphics software.

Output primitives: Points and lines, line drawing algorithms: DDA and Bresenham's line generation, mid-point circle and ellipse algorithms. Filled area primitives: Scan line polygon fill algorithm, boundary-fill and flood-fill algorithms, Fill-Area Functions, Cell Array, Character generation.

UNIT-II

Attributes of Output Primitives: Line Attributes, Curve Attributes, color and gray scale levels, Area Fill Attributes, Character Attributes, Bundled Attributes, Inquiry Functions.

Structures and Hierarchical Modeling: Structure concepts, Editing Structures, Hierarchical modeling with structures. Graphical User Interfaces and Interactive

Input Methods: The User Dialogue, Logical Classification of Input Devices, Input Functions, Interactive Picture Construction Techniques.

UNIT-III

2-D Geometrical transforms: Translation, scaling, rotation, reflection and shear transformations, matrix representations and homogeneous coordinates, composite transforms, transformations between coordinate systems.

2-D Viewing: The viewing pipeline, viewing coordinate reference frame, window to view-port coordinate transformation, viewing functions, Cohen-Sutherland and Liang-Barsky line clipping algorithms, Sutherland –Hodgeman polygon clipping algorithm.

UNIT-IV

3-D Object representation: Polygon surfaces, quadric surfaces, spline representation, Hermite curve, Bezier curve and B-spline curves, Bezier and B-spline surfaces, CSG, Octrees, BSP Trees.

3-D Geometric transformations: Translation, rotation, scaling, reflection and shear transformations, composite transformations, 3-D viewing: Viewing pipeline, viewing coordinates, projections, view volume and general projection transforms.

UNIT-V

Visible surface detection methods: Classification, back-face detection, depth-buffer, scan-line, depth sorting, BSP-tree methods, area sub-division and octree methods.

Computer animation: Design of animation sequence, general computer animation functions, raster animation, computer animation languages, key frame systems, motion specifications.

Text Books:

1. Donald Hearn and M. Pauline Baker, "Computer Graphics C version", Second Edition, Pearson Education.
2. "Computer Graphics Principles & practice", second edition in C, Foley, VanDam, Feiner and Hughes, Pearson Education.

Suggested Reading:

1. "Computer Graphics" Second edition, Zhigandxiang, Roy Plastock, Schaum's outlines, Tata Mc- Graw hill edition.
2. Procedural elements for Computer Graphics, David F Rogers, Tata Mc Graw hill, 2nd edition.
3. "Principles of Interactive Computer Graphics", Neuman and Sproul, TMH.
4. Principles of Computer Graphics, ShaliniGovil, Pai, 2005, Springer.
5. Computer Graphics, Steven Harrington, TMH.

Web Resources:

1. http://nptel.iitm.ac.in/courses/Webcourse-contents/IITDelhi/Computer%20Graphics/csmain.html

16ITE05

FILE STRUCTURES
(Elective-II)

Instruction	3LHours per week
Duration of End Examination	3 Hours
Semester End Examination	70 Marks
CIE	30 Marks
Credits	3

Course Objectives:

This course is introduced to

1. Understand File operations.
2. Understand UNIX File system.
3. Understand Indexing and Hashing Concepts to organize data in files.
4. Understand B+-Trees to organize files.

Course Outcomes:

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

1. Understand file structures including sequential, indexed, indexed sequential, hashed file structures.
2. Implement file operations including read, write, update and search.
3. Apply object-oriented concepts to design file systems.
4. Understand B^+-trees to implement file systems.
5. Develop and analyse external sorting methods.

Prerequisites:

Programming and problem solving (16CS01), data structures and algorithms (16ITC01), object oriented programming (16ITC03).

UNIT-I

Fundamental File Structure Concepts: Field and Record Organization, Using Classes to Manipulate Buffers, Using Inheritance for Record Buffer Classes, Managing Fixed Length, Fixed Field Buffers, An Object-Oriented Class for Record Files.

Managing Files and Records: Record Access, More about Record Structures, Encapsulating Record I/O Operations in a Single Class, File Access and File Organization.

UNIT-II

Fundamental File Processing Operations: Physical Files and Logical Files, Opening Files, Closing Files, Reading and Writing, Seeking, Special Characters in Files, The UNIX Directory Structure, Physical Devices and Logical Files, Physical Devices as Files ,File-related Header Files, UNIX File System Commands.

Indexed Files of Data Objects- Indexing: A Simple Index for Entry-Sequenced File, Template Classes in C++, Object-Oriented support for Indexed, Entry-Sequenced Files of Data Objects, Indexes That Are Too Large to Hold in Memory, Indexing to Provide Access by Multiple Keys, Retrieval Using Combinations of Secondary Keys, Improving the Secondary Index Structure: Inverted Lists, Selective Indexes, Binding.

UNIT-III

Multilevel Indexing and B-Trees: Introduction: The Invention of the B-Tree, Statement of the Problem, Indexing with Binary Search Trees, Multi-level Indexing, A Better Approach to Tree Indexes, B-Trees: Working up from the Bottom, Example of Creating a B-Tree, An Object-Oriented Representation of B-Trees, B-Tree Methods Search, Insert, and Others, B-Tree Nomenclature, Formal Definition of B-Tree Properties, Worst-case Search Depth, Deletion, Merging, and Redistribution, Redistribution during Insertion: A Way to Improve Storage Utilization, B* Trees, Buffering of Pages: Virtual B-Trees, Variable-length Records and Keys.

UNIT-IV

Indexed Sequential File Access and B+ Trees : Indexed Sequential Access, Maintaining a Sequence Set, Adding a Simple Index to the Sequence Set, The Content of the Index: Separators Instead of Keys, The Simple Prefix B+ Tree, Simple Prefix B+ Tree Maintenance, Index Set Block Size, Internal Structure of Index Set Blocks: A Variable-order B-Tree, Loading a Simple Prefix B+ Tree, B+ Trees, B-Trees, B+ Trees, and Simple Prefix B+ Trees in Perspective.

Hashing: Introduction, A Simple Hashing Algorithm, Hashing Functions and Record Distributions, How Much Extra Memory Should Be Used, Collision Resolution by Progressive Overflow, Storing More Than One Record per Address: Buckets, Making Deletions, Other Collision Resolution Techniques, Patterns of Record Access.

UNIT-V

Extendible Hashing: Introduction, How Extendible Hashing Works, Implementation, Deletion, Extendible Hashing Performance, Alternative Approaches, Multi list and Inverted Files, Sorting of Large Files,

External sorting: Secondary storage algorithms.

Text Book:

1. Michael j. Folk, Greg Riccardi, Bill Zoellick; *File Structures: An Object Oriented Approach with C++*, 3/e Pearson Publishers.

Suggested Reading:

1. Horowitz, E., and Sahni.S: Fundamentals of Data structures. Computer Science Press, 1978.
2. Wirth, Nicolaus: Algorithms + Data structures = Programs. Prentice-Hall International, 1975.
3. Knuth, D.: The Art of Computer Programming, Vols. 1-2. Addison-Wesley, 1970-80.

16ITE06

OBJECT ORIENTED SYSTEM DEVELOPMENT USING UML
(Elective-II)

Instruction	3LHours per week
Duration of End Examination	3 Hours
Semester End Examination	70 Marks
CIE	30 Marks
Credits	3

Course Objectives:

This course is introduced to:

1. Acquaint the student with the precise vocabulary and powerful notation used in Unified modeling language.
2. Describe the basic structural modeling concepts in UML.
3. Familiarize students with architectural modeling.
4. Explain the concepts of Unified software development process.
5. Acquaint the students with UML notations and discuss several case studies.

Course Outcomes:

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

1. Understand the precise vocabulary and powerful notation used in Unified modeling language.
2. Provide comprehensive introduction to basic structural modeling in UML.
3. Develop the component and deployment diagrams in architectural modeling.
4. Understand the Unified software development process and apply to UML models.
5. Involve in analysis and design of UML models for various case studies.
6. Relate the applications of Unified process in UML modeling.

Prerequisites:

Object Oriented Programming (16ITC03), Software engineering (16ITC26)

UNIT-I

UML Introduction: Why we Model, Introducing the UML, Elements of UML. Basic Structural Modeling: Classes, Relationships, Common Mechanisms, Diagrams, Class Diagrams.

Advanced Structural Modeling: Advanced Classes, Advanced Relationships, Interfaces, Types and Roles, Packages, Instances, Object Diagrams, Components, Case studies on class diagrams.

UNIT-II

Basic BehavioralModeling: Interactions, Use Cases, Use Case Diagrams, Interaction diagrams, Activity diagrams, Case studies on Use Case diagrams, Interaction diagrams.

Advanced BehavioralModeling: Events and Signals-types of events-internal and external events, State Machines, Processes and Threads, Time and space, State Chart Diagrams, Case studies on State chart diagrams.

UNIT-III

Architectural Modeling: Artifacts, Deployment , Collaborations, Patterns and Frame-works, Artifact Diagrams, Deployment Diagrams, components of deployment diagrams-nodes and links, common modeling techniques for deployment diagrams-modeling a fully distributed system, modeling embedded systems, modeling client-server systems, Systems and Models- subsystems, trace relationships, Case studies on Deployment diagrams.

UNIT-IV

Unified Software Development Process: The Unified Process, phases in unified software development process-inception, elaboration, construction and transition, The Four P's-people, project, product, process, A Use-Case Driven Process-Importance of Use case modeling, An Architecture-Centric Processes, base lining the architecture, An Iterative and Incremental Process-a generic iteration, advantages of iterative and incremental process.

UNIT-V

Core Workflows: Requirements Capture, Capturing Requirements as Use Cases, Analysis-role of analysis in software life cycle, artifacts, workers and activities in analysis workflow, Design-workers, artifacts and activities in design workflow, Implementation-role of implementation in software life cycle, Test, testing artifacts-test case, test plan, test procedure.

Text Books:

1. Grady Booch, James Rumbaugh, Ivor Jacobson, "The Unified Modeling Language-User Guide (Covering UML 2.0)", Third Edition, Pearson Education, India, 2010.
2. Ivor Jacobson, Grady Booch, James Rumbaugh, "The Unified Software Development Process", second edition ,Pearson Education, India, 2008.

Suggested Reading:

1. Martin Fowler, Kendall Scott "UML Distilled**:** A Brief Guide to the Standard Object Modeling Language" Addison Wesley, Fourth Edition, 2011.
2. Hans van Vliet "Software Engineering Principles and Practice", Second Edition, 2010.

Web Resources:

1. IBM Rational http://www-306.ibm.com/software/rational/uml/
2. Practical UML - A Hands-On Introduction for Developers http://www.togethersoft.com/services/practical_guides/umlonlinecourse/
3. http://www-inst.eecs.berkeley.edu/~cs169/

16ITE07

DIGITAL IMAGE PROCESSING
(Elective-III)

Instruction	3LHours per week
Duration of End Examination	3 Hours
Semester End Examination	70 Marks
CIE	30 Marks
Credits	3

Course Objectives:

1. To learn the fundamental concepts and applications of digital image processing.
2. To learn the image processing concepts: intensity transformations, spatial filtering, smoothing and sharpening in both spatial and frequency domains, image restoration and reconstruction.
3. To learn the image analysis concepts: morphological image processing, image segmentation, image representation and description, and object recognition.
4. To understand colour image processing techniques.
5. To learn various image compression methods.

Course Outcomes:

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

1. Explain the fundamental concepts and discuss the applications of digital image processing.
2. Explain intensity transformations, spatial filtering, smoothing and sharpening in both spatial and frequency domains, image restoration and reconstruction.
3. Demonstrate the image analysis concepts like morphological image processing, image segmentation, image representation and description, and object recognition.
4. Illustrate colour image processing techniques.
5. Distinguish and describe various image compression methods.

Prerequisites:

Engineering Mathematics- I (16MTCO1)

UNIT-I

Basics: Introduction, Fundamental Steps in Digital Image Processing, Components of an Image Processing System, Elements of visual perception, Image Sampling

and Quantization - Basic Concepts in Sampling and Quantization, Representing Digital Images, Spatial and Intensity Resolution;

Some Basic Relationships between Pixels - Neighbours of a Pixel, Adjacency, Connectivity, Regions, and Boundaries, Distance Measures
Intensity Transformations: Some Basic Intensity Transformation Functions, Image Negatives, Log Transformations, Power-Law (Gamma) Transformations, Piecewise-Linear Transformation Functions, Histogram Processing - Histogram Equalization, Histogram Matching (Specification), Local Histogram Processing.

UNIT- II

Spatial Filtering: Fundamentals of Spatial Filtering, The Mechanics of Spatial Filtering, Spatial Correlation and Convolution, Smoothing Spatial Filters - Smoothing Linear Filters, Order-Statistic (Nonlinear) Filters; Sharpening Spatial Filters – Foundation, Using the Second Derivative for Image Sharpening—The Laplacian, Unsharp Masking and Highboost Filtering.

Filtering in the Frequency Domain: The 2-D Discrete Fourier Transform and its inverse, Some Properties of the 2-D Discrete Fourier Transform - Relationships Between Spatial and Frequency Intervals, Translation and Rotation, Periodicity, Symmetry Properties, Fourier Spectrum and Phase Angle, The 2-D Convolution Theorem.

The Basics of Filtering in the Frequency Domain - Frequency Domain Filtering Fundamentals Correspondence Between Filtering in the Spatial and Frequency Domains, Image Smoothing Using Frequency Domain Filters, Ideal Low pass Filters, Butterworth Low pass Filters, Gaussian Low pass Filters, Image Sharpening Using Frequency Domain Filters - Ideal High pass Filters, Butterworth High pass Filters, Gaussian High pass Filters, The Laplacian in the Frequency Domain, Unsharp Masking, Highboost Filtering.

UNIT- III

Image Restoration and Reconstruction: A Model of the Image Degradation/ Restoration Process, Noise Models - Spatial and Frequency Properties of Noise, Some Important Noise Probability Density Functions, Periodic Noise, Estimation of Noise Parameters, Restoration in the Presence of Noise Only—Spatial Filtering, Mean Filters, Order-Statistic Filters, Adaptive Filters; Periodic Noise Reduction by Frequency Domain Filtering – Band reject Filters, Band pass Filters; Estimating the Degradation Function - Estimation by Image Observation, Estimation by Experimentation, Estimation by Modelling; Inverse Filtering; Minimum Mean Square Error (Wiener) Filtering; Constrained Least Squares Filtering.

Morphological Image Processing: Preliminaries, Erosion and Dilation, Opening and Closing.

UNIT- IV

Image Segmentation:Fundamentals, detection of isolated points, line detection, basic edge detection, edge linking and boundary detection; thresholding – foundation, basic global thresholding, optimum global thresholding using otsu's method; region-based segmentation - region growing, region splitting and merging; segmentation using morphological watersheds - background, dam construction, watershed segmentation algorithm.

Representation and Description: Representation-Boundary (Border) Following, Chain Codes, Polygonal Approximations Using Minimum-Perimeter Polygons, Signatures, Boundary Descriptors - Some Simple Descriptors, Shape Numbers, Fourier Descriptors, Statistical Moments, Regional Descriptors - Some Simple Descriptors, Topological Descriptors, Texture.

Object Recognition: Patterns and Pattern Classes, Recognition Based on Decision-Theoretic Methods – Matching, Optimum Statistical Classifiers, Neural Networks.

UNIT-V

Colour Image Processing: Colour Fundamentals; Colour Models - RGB Colour Model, CMY and CMYK Colour Models, The HSI Colour Model; Pseudo colour Image Processing - Intensity Slicing, Intensity to Colour Transformations; Basics of Full-Colour Image Processing - Colour Transformations, Colour Edge Detection
Image Compression: Fundamentals-Coding Redundancy, Spatial and Temporal Redundancy, Irrelevant Information, Measuring Image Information, Fidelity Criteria, Image Compression Models - Image Formats, Containers, and Compression Standards; Some Basic Compression Methods - Huffman Coding, Arithmetic Coding, LZW Coding, Block Transform Coding.

Text Book:

1. Rafael C Gonzalez and Richard E Woods, "Digital Image Processing", Pearson Education, Third Edition.

Suggested Reading:

1. Vipula Singh, "Digital Image Processing with MatLab and lab View", Elsevier.
2. Thomas B. Moeslund, "Introduction to Video and Image Processing: Building Real Systems and Applications", Springer, 2012.
3. Milan Sonka, Vaclav Halvac and Roger Boyle, "Image Processing, Analysis, and Machine Vision", Second Edition, Thomson Learning Publishers.
4. Kenneth R.Castleman, "Digital Image Processing", Pearson Education, 2006.

16ITE08

INFORMATION RETRIEVAL SYSTEMS
(Elective – III)

Instruction	3LHours per week
Duration of End Examination	3 Hours
Semester End Examination	70 Marks
CIE	30 Marks
Credits	3

Course Objectives:

This course is introduced to

1. To familiarize the different Information Retrieval models.
2. To understand how to write query languages and evaluation.
3. To build index and perform compression on the data.
4. To familiarize pattern matching algorithms.
5. To learn parallel and distributes models.

Course Outcomes:

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

1. Understand different Information Retrieval models.
2. Understand the query language to retrieve the data.
3. Analyse and improve the retrieval results.
4. Understands the operations on the text data and builds index of the data.
5. Apply different pattern matching algorithms on text data.
6. Understand parallel and distributed Information Retrieval models.

Prerequisites:

Database Systems (16ITC17), Data Warehousing and Data Mining (16ITC25).

UNIT-I

Introduction: Basic concepts, Past present and Future of IRS, Retrieval Process.
Modeling: Introduction, A Taxonomy of IR Models, Retrieval: Adhoc and Filterig, A formal characterization of IR Models, Classic IR, Set Theoretic Models, Algebraic Models, Probabilistic Models.

UNIT-II

Structured Text Retrieval Models, Models for Browsing.
Retrieval Evaluation: Introduction, Reference Collections.
Query languages: Introduction, Keyword-based querying, pattern Matching, Structural Queries, Query Protocols.

UNIT-III

Query operations: Introduction, User Relevance Feedback, Automatic Local Analysis, Automatic Global Analysis.

Text and Multimedia Languages and Properties: Introduction, Meta Data, Text, Markup Languages, Multimedia.

UNIT-IV

Text operations: Introduction, Document Preprocessing, Document Clustering, Text Compression, Comparing Text Compression Techniques Indexing: Introduction, Inverted Files, Other Indices for Text Searching, Boolean Queries.

UNIT-V

Searching: Sequential Searching, Pattern Matching, Structural Queries, Compression Parallel and Distributed IR: Introduction, Parallel IR, Distributed IR.

Text book:

1. Ricardo, Baeza-yates, BerthierRibeiro-Neto, "Modern Information Retrieval" Pearson Education, 2008.

Suggested Reading:

1. Christopher D. Manning and PrabhakarRaghavan and HinrichSchütze, "Introduction to
2. Information Retrieval", Cambridge University Press, 2009.
3. David A. Grossman, OphirFrieder, "Information Retrieval - Algorithms and Heuristics", Springer, 2nd Edition (Distributed by Universities Press), 2004.
4. Gerald Kowalski, "Information Retrieval Systems: Theory and Implementation", Springer.
5. William B. Frakes, Ricardo Baeza- Yates, "Information Retrieval – Data Structures & Algorithms", Pearson Education, 2008.

Web Resources:

1. https://class.coursera.org/nlp/lecture
2. http://www.dcs.gla.ac.uk/Keith/Preface.html

16ITE09

E-COMMERCE
(Elective-III)

Instruction	3LHours per week
Duration of End Examination	3 Hours
Semester End Examination	70 Marks
CIE	30 Marks
Credits	3

Course Objectives:

This course is introduced to

1. Analyze features of existing e-commerce businesses, and propose future directions or innovations for specific businesses.
2. To understand the role of multimedia in E-Commerce and security issues of E-Commerce.
3. Discuss electronic commerce and the stakeholders and their capabilities and limitations in the strategic convergence of technology and business.
4. Identify advantages and disadvantages of technology choices such as merchant server software and electronic payment options.
5. To understand the Emerging tools for Resource search and discovery.

Course Outcomes:

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

1. Able to implement e-commerce in business applications.
2. To make effective use of multimedia in E-commerce applications.
3. To resolve security issues in Electronic Payment Systems.
4. Able to describe the Document infrastructure for E-commerce and advertisement in Market.
5. To make use of emerging tools in Resource search and discovery.
6. Be aware of global perspectives (needs, rules/regulations, and specifications).

UNIT-I

Introduction: Electronic commerce and Physical Commerce, different type of ecommerce, some e-commerce scenario, Advantages of e-commerce.

Basic technologies of Ecommerce: Client side Programming, Server Side Programming, Database connectivity, session tracking techniques.

UNIT-II

Internet Payment System: Characteristics of payment system, SET Protocol for creditcard payment, E-cash, E-check, Micropayment system.

E-commerce strategies: Strategies for marketing, Sales and Promotions, Strategies forPurchasing and support activities, Strategies for Web Auctions, Virtual Communities,and web portals.

UNIT -III

E-Business -Introduction: E-Business vs E-commerce,, Characteristics of e-Business, e-Business role and their challenges, e-business Requirements, impacts of e-business.

E-business strategies: Strategic positioning, Levels of e-business strategies, Strategicplanning process, Strategic alignment, the consequences of e-Business, Success factorsfor implementation of e-business strategies. Business models, Business process and collaborations.

UNIT-IV

Advance technologies of E-commerce: Mobile Agent, WAP, XML, Data Mining, RichInternet Application, Web 2.0, REST Web Services, Web Mashup, Working of SearchEngines, Internet Security.

UNIT- V

Integration of Application: Approaches to Middleware, RPC and RMI, EnterpriseApplication Integration, e-business Integration, loosely Coupled e-Business solutions forintegration, Service Oriented Architecture, EAI and web Services, WS-security.

Text Books:

1. E-Commerce Fundamentals and application (Henry Chan) Wiley publication.
2. Electronic Commerce (Gary Schneider) Thomson Course technology.
3. E-Business Organizational and technical foundation (Michael P) Wiley Publication.

Suggested Reading:

1. E- Commerce Strategies, Technology and applications (David) Tata McGraw-Hill.
2. Introduction to E-commerce (Jeffrey) Tata- McGraw-Hill.
3. E-Business and Commerce- Strategic Thinking and Practice (Brahm) biztantra.

Web Resources:

1. http://www.w3schools.com/xml/default.asp
2. http://www.tizag.com/xmlTutorial/
3. https://www.practicalecommerce.com/

16ITC28

NETWORK PROGRAMMING LAB

Instruction	3LHours per week
Duration of End Examination	3 Hours
Semester End Examination	50 Marks
CIE	25 Marks
Credits	2

Course Objectives:

This course is introduced to

1. Familiarize students with client/server architecture in application development.
2. Provide understanding of elementary socket system calls, advanced socket system calls.
3. Expose students to the usage of TCP and UDP based sockets.
4. Provide knowledge of network routing algorithms and application layer protocols.
5. Cryptographic principles and encryption algorithms.

Course Outcomes:

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

1. Use elementary socket system calls and develop distributed applications.
2. Model and evaluate performance of networking systems.
3. Implement the Routing algorithms.
4. Develop and implement next generation protocols required for emerging applications.
5. Understand the operating principles of Electronic Mail (SMTP), HTTP.
6. Comprehend various network security threats and implement the cryptographic algorithms.

Prerequisites:

Programming and Problem Solving (16CSC01), Java Programming (16ITC10).

List of Programs

1. Understanding and using of commands like ifconfig, netstat, ping, arp, telnet, ftp, finger, traceroute, whoisetc. Usage of elementary socket system calls (socket(), bind(), listen(), accept(), connect(), send(), recv(), sendto(), recvfrom()).
2. Implementation of Connection oriented concurrent service (TCP).
3. Implementation of Connectionless Iterative time service (UDP).
4. Implementation of Select system call.
5. Implementation of getsockopt(), setsockopt() system calls.

6. Implementation of getpeername() system call.
7. Implementation of remote command execution using socket system calls.
8. Implementation of Distance Vector Routing Algorithm.
9. Implementation of HTTP.
10. Implementation of RSA algorithm.
11. Develop an Internet Mail Application.
12. Multimedia file transmission using FTP.

Note: Implement programs 2 to 7 in C and 8 to 12 in JAVA.

Text Book:

1. W. Richard Stevens, "Unix Network Programming", Prentice Hall, Pearson Education, 2009.

Suggested Reading:

1. Douglas E.Comer, "Hands-on Networking with Internet Technologies", Pearson Education.
2. James Kurose and Keith Ross. Computer Networking: A Top-Down Approach Featuring the Internet.

Web Resources:

1. https://in.udacity.com/course/computer-networking—ud436
2. https://www.mooc-list.com/course/learn-socket-programming-tutorial-c-scratch-eduonix.

16ITC29

DATA MINING LAB

Instruction	3LHours per week
Duration of End Examination	3 Hours
Semester End Examination	50 Marks
CIE	25 Marks
Credits	2

Course Objectives:

This course is introduced to

1. Weka tool and R-Tool for data mining.
2. Present various pre-processing techniques.
3. Familiarise with data visualization.
4. Acquaint various features available in weka for mining interesting patterns.
5. Present various mining techniques to analyse the data in R - Tool.

Course Outcomes:

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

1. Describe the data using various visualisation techniques.
2. Identify and apply necessary pre-processing techniques on raw data.
3. Generate interesting patterns using appropriate data mining techniques.
4. Perform pattern evaluation.
5. Visualise the knowledge mined.
6. Build a data mining system for a given application.

Prerequisites:

Database Systems (16ITC17)

List of Programs

I. Introduction to data mining using Weka and R-Tool.
II. Experiment the following in Weka Tool.
    1. Perform the following Preprocessing operations:
        i. Attribute selection
        ii. Handling missing values
        iii. Discretisation
        iv. Converting nominal attributes to binary attributes
        v. Normalisation
        vi. Standardisation
        vii. Outlier detection and elimination.

2. Generate Association Rules using Apriori and FP Growth algorithms.
3. Build the following classifiers and check their efficiency:
    i. Decision Tree
    ii. Naïve Bayes
    iii. Bagging
    iv. AdaBoost
    v. Random forest
    vi. K-NN
4. Apply the following clustering algorithms on datasets and visualise the clusters
    i. K-Means
    ii. Hierarchical
    iii. DBSCAN
5. Build Linear Regression model.

III. Experiment the following in R-Tool:
1. Data Import/Export
2. Data Exploration and Visualization
3. Association Rule Mining
4. Regression and Classification
5. Data Clustering
6. Text Mining with R: Twitter Data Analysis
7. Time Series Analysis and Mining

(Note: Wherever necessary interpret the results and measure the performance)

Text Books:

1. Ian H.Witten, EibeFank, Mark A Hall, "Data Mining Practical Machine Learning Tools and Techniques", Third edition, 2011.
2. Pawel Cichosz, "Data Mining Algorithms: Explained Using R", Wiley (2015).

Suggested Reading:

1. Han J, Kamber M, Jian P "Data Mining: Concepts and Techniques", Third Edition, Elsevier, 2012.
2. Yanchang Zhao, "R and Data mining: Examples and Case Studies", First Edition, Elsevier 2012.

Web Resources:

1. https://www.cs.waikato.ac.nz/ml/weka/
2. http://www.rdatamining.com/
3. http://illimine.cs.uiuc.edu/
4. https://www.kdnuggets.com/
5. http://archive.ics.uci.edu/ml/index.php

16ITC30

MINI PROJECT – IV

Instruction	2 Hours per week
Duration of End Examination	-
Semester End Examination	-
CIE	50 Marks
Credits	1

Course Objectives:

1. To enable students learn by doing.
2. To develop capability to analyze and solve real world problems.
3. To develop innovative ideas among the students.

Course Outcomes:

Students should be able to do the following:

1. To provide innovative solutions.
2. To work in a team.
3. To manage time and resources in the best possible manner.

The Students are required to implement one of the projects from project exercise given in the suggested readings of the theory subjects of the current semester / as suggested by the respective course faculty of that semester. During the implementation of the project, Personnel Software Process (PSP) has to be followed. Report of the project work has to be submitted for evaluation.

CHAITANYA BHARATHI INSTITUTE OF TECHNOLOGY (A)

Choice Based Credit System (with effect from 2019-20)

B.E (Information Technology)

PROPOSED SCHEME OF INSTRUCTION AND EXAMINATION

Semester– VII

S.No	Syllabus Ref. No	Subject	Scheme of Instruction		Scheme of Examination			Credits
			Hours per Week		Duration of SEE in Hours	Maximum Marks		
			L/T	P/D		CIE	SEE	
THEORY								
1	16ITC31	Embedded Systems &IoT	3	-	3	30	70	3
2	16ITC32	Distributed Systems	3	-	3	30	70	3
3	16ITC33	Information Security	3	-	3	30	70	3
4	16ITC34	Big Data Analytics	3	-	3	30	70	3
5		Elective -IV	3	-	3	30	70	3
6		Open Elective - I	3	-	3	30	70	3
PRACTICAL								
7	16ITC35	Big Data Analytics Lab	-	3	3	25	50	2
8	16ITC36	Embedded Systems &IoT Lab	-	3	3	25	50	2
9	16ITC37	Project Seminar	-	3	-	50	-	2
		TOTAL	18	9	-	280	520	24

L: Lecture T: Tutorial D: Drawing P: Practical
CIE-Continuous Internal Evaluation SEE-Semester End Examination

Elective-IV

S.No.	Subject Code	Subject Name
1.	16ITE10	Cloud Computing
2.	16ITE11	Soft Computing
3.	16ITE12	VLSI Technology

Open Elective - I

S.No.	Subject Code	Subject Name
1.	16XXXX	Research Methodologies
2.	16XXXX	Intellectual Property Rights & Cyber Laws
3.	16XXXX	Operations Research

CHAITANYA BHARATHI INSTITUTE OF TECHNOLOGY (A)

Choice Based Credit System (with effect from 2019-20)

B.E (Information Technology)

PROPOSED SCHEME OF INSTRUCTION AND EXAMINATION

Semester– VIII

S.No	Syllabus Ref. No	Subject	Scheme of Instruction		Scheme of Examination			Credits
			Hours per Week		Duration of SEE in Hours	Maximum Marks		
			L/T	P/D		CIE	SEE	
THEORY								
1		Elective -V	3	-	3	30	70	3
2		Elective - VI	3	-	3	30	70	3
3		Open Elective - II	3	-	3	30	70	3
4	16ITC38	Seminar	3	-	-	50	-	2
5	16ITC39	Project	-	6	3 (viva)	50	100	6
		TOTAL	12	6	-	190	310	17

L: Lecture **T: Tutorial** **D: Drawing** **P: Practical**
CIE-Continuous Internal Evaluation **SEE-Semester End Examination**

Elective-V

S.No.	Subject Code	Subject Name
1.	16ITE13	Virtual and Augmented Reality
2.	16ITE14	Social Media Analytics
3.	16ITE15	Human Computer Interaction

Elective-VI

S.No.	Subject Code	Subject Name
1.	16ITE16	Natural Language Processing
2.	16ITE17	Mobile Computing
3.	16ITE18	Business Intelligence

Open Elective - II

S.No.	Subject Code	Subject Name
1.	16XXXX	Entrepreneurship
2.	16XXXX	Organizational Behaviour
3.	16XXXX	Disaster Mitigation and Management

16ITE25

JAVA PROGRAMMING
(Elective)
(Service Course offered to ECE Program)

Instruction	3L Hours per week
Duration of End Examination	3 Hours
Semester End Examination	70 Marks
CIE	30 Marks
Credits	3

Course Objectives:

1. To understand the fundamentals of Java language which includes defining classes, invoking methods, inheritance, polymorphism, exception handling etc.
2. To solve real world problems by creating Java applications using sound OOP practices, standard class libraries and APIs.
3. To introduce event driven Graphical User Interface (GUI) programming and usage ofstandard class libraries.

Course Outcomes:

1. Achieve proficiency in object-oriented concepts and also learns to incorporate the same into the Java programming language.
2. Create Java application programs using sound OOP practices e.g. Inheritance, interfaces and proper program structuring by using packages, access control specifiers.
3. Understand and Implement the concepts of Exception Handling and Multithreading in java.
4. Develop the ability to solve real-world problems through software development in high-level programming language using Large APIs of Java as well as the Java standard class library.
5. Understand File, Streams, Input and Output Handling in java.
6. Create graphical user interfaces in java as well as apply the knowledge of Event Handling.

Prerequisites:
Programming and Problem Solving (16CSC01)

UNIT-I
Evolution of java: Java's Magic: The Bytecode, The Java Buzzwords Objects, **Overview of Java:** Simple Java Programs, Java Primitive Types, Arrays: How to create and define arrays, Basic Operators, Control statements.
Introducing Classes: Declaring objects, methods, Constructors, this keyword, Method Overloading and Constructor Overloading, Objects as parameters, Returning objects, Use of static and final keywords.

UNIT-II

Inheritance, Packages and Interfaces: Inheritance basics, using super keyword**,** Method overriding, Dynamic method dispatch , Abstract classes, using final with inheritance , Introduction to Object class, **Packages:** Defining, Creating and Accessing a Package, importing packages, **Interfaces :** Defining and implementing interfaces, Nested Interfaces.

Strings Handling: String Constructors, Length, Operations, String Comparison, Searching for strings, Difference between String &StringBuffer classes, StringTokenizer class and Wrapper classes and conversion between Objects and primitives.

UNIT-III

Exception Handling in Java: Exception handling fundamentals, Exception types**,** Usage of try, catch, throw, throws and finally clauses, writing your own exception classes.

Multithreading in Java: The java Thread Model, How to create threads, Thread class in java, Thread priorities, Thread synchronization.

Generics: What are Generics? Generic classes, bounded types, Generic methods and interfaces.

UNIT-IV

Collections Framework: Overview of Collection Framework, Commonly used Collection classes – ArrayList, LinkedList, HashSet, LinkedHashSet, TreeSet, Collection Interfaces –Collection, List, Set, SortedSet, Accessing a collection via an Iteration,Storing user-defined classes in collections, Map Interfaces and Classes, Using a comparator. Legacy classes – Vector, Hashtable, The Enumeration interface.

Input/Output : How to read user input (from keyboard) using scanner class,Stream classes, InputStream, OutputStream, FileInputStream, FileOutputStream, Reader and Writer, FileReader, FileWriter cl asses. File class.

UNIT-V

GUI Design & Event Handling: Component, Container, window, Frame classes. Working with Frame windowGUI Controls, Layout Managers, Introduction to Swings, Delegation Event Model, Event Classes, Source of Events, Event Listener Interfaces, Handling button click events, Adapter classes.Writing GUI Based applications. **Database Handling in Java**: Java Database Connectivity (JDBC)

Text Books:

1. Herbert Schildt: "Java: The Complete Reference", 8th Edition, Tata McGraw Hill Publications, 2011.
2. Cay S. Horstmann, Gary Cornell: "Core Java, Volume I—Fundamentals", 8th edition, Prentice Hall, 2008.

Suggested Reading:

1. Sachin Malhotra & Saurabh Choudhary: "Programming in Java", 2nd Edition, Oxford University Press, 2014.
2. C. Thomas Wu, "An introduction to Object-oriented programming with Java", 4th edition, Tata McGraw-Hill Publishing company Ltd., 2010.
3. K. Arnold and J. Gosling, "The JAVA programming language", 3rdedition, Pearson Education, 2000.

Web Resources:

1. https://www.cse.iitb.ac.in/~nlp-ai/javalect_august2004.html.
2. http://nptel.ac.in/courses/106106147/
3. https://ocw.mit.edu/courses/electrical-engineering-and-computer-science/6-092-introduction-to-programming-in-java-january-iap-2010/lecture-notes/

16ITE26

PYTHON PROGRAMMING
(Elective)
(Service Course offered to ECE Program)

Instruction	3L Hours per week
Duration of End Examination	3 Hours
Semester End Examination	70 Marks
CIE	30 Marks
Credits	3

Course Objectives:
This course is introduced to
1. Introduce the fundamentals of Python programming
2. Learn how to use lists, tuples, and dictionaries in Python programs.
3. Learn how to read and write files in Python.
4. Impart usage of exception handling for error handling.
5. Familiarize python visualization.

Course Outcomes:
After completion of the course, student will be able to:
1. Understand basic data structures of python.
2. Perform operations on strings.
3. Understand the concepts of file I/O.
4. Understand exception handling in Python.
5. Plot data using appropriate Python visualization libraries.
6. Develop basic Python applications.

Prerequisites:
Programming and Problem Solving (16CSC01), Programming Laboratory (16CSC02).

Unit-I
Introduction to Python Programming: Using Python, The IDLE Programming Environment, Input and Output Processing, Displaying Output with the Print Function, Comments, Variables, Reading Input from the Keyboard, Performing Calculations, More About Data Output: New line, Item Separator, Escape Characters, Formatting parameters.
Decision Structures and Boolean Logic: if, if-else, if-elif-else Statements, Nested Decision Structures, Comparing Strings, Logical Operators, Boolean Variables.

Unit-II
Repetition Structures: Introduction, while loop, for loop, Sentinels, Input Validation Loops, Nested Loops.

Functions: Introduction, Defining and Calling a Function, Designing a Program to Use Functions, Local Variables, Passing Arguments to Functions, Global Variables and Global Constants, Value-Returning Functions Generating Random Numbers, Writing Our Own Value-Returning Functions, The math Module, Random Module, Time Module and Storing Functions in Modules.

Unit-III

Lists and Tuples: Sequences, Introduction to Lists, List slicing, Finding Items in Lists with the in Operator, List Methods and Useful Built-in Functions, Copying Lists, Processing Lists, Two-Dimensional Lists, Tuples.

Strings: Basic String Operations, String Slicing, Testing, Searching, and Manipulating Strings.

Unit-IV

Dictionaries and Sets: Dictionaries, Sets, Serializing Objects.

Recursion: Introduction, Problem Solving with Recursion, Examples of Recursive Algorithms.

Python File Input-Output: Opening and closing file, various types of file modes, reading and writing to files, manipulating directories.

Unit-V

Exception Handling: What is exception, various keywords to handle exception such try, catch, except, else, finally, raise.

Regular Expressions: Concept of regular expression, various types of regular expressions, using match function.

Introduction to plotting in Python – Basic Plots- Line and Scatter Plot, Histograms and plotting data contained in files.

Text Books:

1. Tony Gaddis, "Starting Out With Python", 3rd edition, Pearson, 2015.
2. Charles Dierbach, "Introduction to Computer Science using Python", Wiley, 2013.

Suggested Reading:

1. Kenneth A. Lambert, "Fundamentals of Python", Delmar Cengage Learning, 2013.
2. James Payne, "Beginning Python using Python 2.6 and Python 3", wrox programmer to programmer, 2010.
3. Paul Gries, "Practical Programming: An Introduction to Computer Science using Python", 3rd edition, 2016.
4. Clinton W. Brownley, "Foundations for Analytics with Python", 1st edition, O'Rielly Media, 2016.

Web Resources:

1. https://www.python.org/
2. https://www.coursera.org/learn/python
3. https://learnpythonthehardway.org/book/
4. https://www.coursera.org/specializations/python

16ITE27

DATA STRUCTURES
(Elective)
(Service Course offered to ECE Program)

Instruction	3L Hours per week
Duration of End Examination	3 Hours
Semester End Examination	70 Marks
CIE	30 Marks
Credits	3

Course Objectives:

1. To familiarize with different linear and nonlinear data structures.
2. To present the concepts of time and space complexity.
3. To discuss applications of various data structures.
4. To develop a base for advanced computer science study.

Course Outcomes:

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

1. Understand basic data structures arrays and linked lists.
2. Analyse time complexity of algorithms.
3. Understand the basic operations of Stacks and Queues.
4. Implement basic operations on data structures.
5. Understand applications of binary trees and graphs.
6. Understand various kinds of searching and sorting techniques.

Prerequisites:

Programming and Problem Solving (16CSC01), Programming Laboratory (16CSC02)

UNIT-I

Introduction to Data Structures and Algorithms: Elementary data structure organisation, classification of data structures, operations on data structures, Abstract Data Type, Algorithms, Different approaches to designing an algorithm, Control structures used in algorithms, Time and Space Complexity, Big O Notation, Omega Notation (Ω), Theta Notation (Θ)

UNIT-II

Arrays: Introduction, Declaration of Arrays, Accessing the Elements of an Array, Storing Values in Arrays, Operations on Arrays, **Linked Lists:** Introduction, Singly Linked Lists, Circular Linked Lists, Doubly Linked Lists, Applications of Linked Lists

UNIT-III

Stacks: Introduction to Stacks, Array Representation of Stacks, Operations on a Stack, Linked Representation of Stacks, Operations on a Linked Stack, Applications of Stacks, **Queues:** Introduction to Queues, Array Representation of Queues, Linked Representation of Queues, Types of Queues, Applications of Queues.

UNIT-IV

Trees: Introduction, Types of Trees, Creating a Binary Tree from a General Tree, Traversing a Binary Tree, Applications of Trees, **Efficient Binary Trees:** Binary Search Trees, Operations on Binary Search Trees.

UNIT-V

Graphs: Introduction, Graph Terminology, Directed Graphs, Bi-connected Components, Representation of Graphs, Graph Traversal Algorithms **Introduction to Searching:** Linear Search, Binary Search, Introduction to Sorting, Bubble Sort, Insertion Sort, Selection Sort, Merge Sort, Quick Sort, Radix Sort, Heap Sort, Shell Sort, Tree Sort, Comparison of Sorting Algorithms.

Text Books:

1. ReemaThareja, "Data Structures Using C", Second Edition, Oxford Higher Education, 2014.
2. Horowitz Ellis, SahniSartaj& Anderson-Freed Susan, "Fundamentals of Data Structures in C", Orient BlackSwan, 2008.

Suggested Reading:

1. NarasimhaKarumanchi, "Data Structures and Algorithms Made Easy: Data Structures and Algorithmic Puzzles", CareerMonk Publications, 2016.
2. NarasimhaKarumanchi, "Coding Interview Questions", 3rd Edition, CareerMonk Publications, 2016.
3. Yashavant P. Kanetkar, "Data Structure Through C", BPB Publications, 2003.

Web Resources:

1. NPTEL Videos Introduction to data structures and algorithms - http://nptel.ac.in/courses/106102064/1
2. https://www.cs.usfca.edu/~galles/visualization/Algorithms.html
3. https://visualgo.net/en