RASHTRASANT TUKADOJI MAHARAJ NAGPUR UNIVERSITY, NAGPUR
FOUR YEAR BACHELOR OF TECHNOLOGY (B.TECH.) DEGREE COURSE
SEMESTER: SEVENTH (CBCS)
BRANCH: Computer Engineering
Subject : Cryptography and Network Security

Subject : Cryptography and Network Security **Subject Code BTCME701T**

Load	Credit	Total Marks	Internal Marks	University Marks	Total
04Hrs (Theory)	03(L)+01(T)	100	30	70	100

Aim :To highlight the features of different technologies involved in Network Security.

Prerequisite(s): Mathematics, Algorithm, Networking

Course Objectives:

1	To develop the student's ability to understand the concept of security goals in various applications and learn classical encryption techniques
2	Apply fundamental knowledge on cryptographic mathematics used in various symmetric and asymmetric key cryptography
3	To develop the student's ability to analyze the cryptographic algorithms.
4	To develop the student's ability to analyze the cryptographic algorithms.

Course Outcomes:
At the end of this course student are able to:

CO1	To understand basics of Cryptography and Network Security and classify the symmetric encryption techniques.
CO2	Understand, analyze and implement the symmetric key algorithm for secure transmission of data.
CO3	Acquire fundamental knowledge about the background of mathematics of asymmetric key cryptography and understand and analyze asymmetric key encryption algorithms and digital signatures.
CO4	Analyze the concept of message integrity and the algorithms for checking the integrity of data.
CO5	To understand various protocols for network security to protect against the threats in the networks.

UNIT-I **[08 Hrs]**

Introduction, Model for network security. Mathematics of cryptography: modular arithmetic, Euclidean and extended Euclidean algorithm. Classical encryption techniques: substitution techniques-Caesar cipher, Vigenere's ciphers, Playfair ciphers and transposition techniques.

UNIT-II **[07 Hrs]**

Symmetric key cryptography: Block Cipher Principles, Data Encryption Standard (DES), Triple DES, Advanced Encryption Standard (AES), RC4, Key Distribution.

UNIT III **[07 Hrs]**

Asymmetric key cryptography: Euler's Totient Function, Fermat's and Euler's Theorem, Chinese Remainder Theorem, RSA, Diffie Hellman Key Exchange, ECC, Entity authentication: Digital signature.

UNIT IV **[07 Hrs]**

Message Integrity and authentication: Authentication Requirements and Functions, Hash Functions, MD5, Kerberos, Key Management, X.509 Digital Certificate format.

UNIT V **[07 Hrs]**

Network Security: PGP, SSL, Firewalls, IDS, Software Vulnerability: Phishing, Buffer Overflow, SQL Injection, Electronic Payment Types

Text Book:

1. William Stallings, "Cryptography and Network Security: Principles and Standards", Prentice Hall India, 7th Edition, 2017.
2. Bernard Menezes, "Network Security and Cryptography", Cengage Learning, 2010.

Reference Books:

1. Robert Bragge, Mark Rhodes, Heithstraggberg "Network Security, The Complete Reference", Tata McGraw Hill Publication, 2004.
2. Behrouz A. Forouzan, "Cryptography and Network Security", McGraw-Hill publication, 2nd Edition, 2010
3. Bruce Schneier, Applied Cryptography, John Wiley New York, 2nd Edition, 1996.

Subject : Cryptography and Network Security **Subject Code BTCME701P**

Load	Credit	Total Marks	Internal Marks	University Marks	Total
02Hrs (Practical)	01	50	25	25	50

Aim : To highlight the features of different technologies involved in Network Security.

Prerequisite(s): Mathematics, Algorithm

Course Objective:

1	To develop the student's ability to understand the concept of security goals in various applications.
2	To provide the students with some fundamental cryptographic mathematics used in various symmetric and asymmetric key cryptography.
3	To develop the student's ability to analyze the cryptographic algorithms.
4	To familiarize the student the need of security in computer networks.

Course Outcome:

At the end of this course student are able to:

CO1	Acquire knowledge about security goals, background of cryptographic mathematics and identification of its application
CO2	Understand, analyze and implement – the symmetric key algorithm
CO3	Acquire knowledge about the background of mathematics of asymmetric key cryptography and understand and analyze – asymmetric key encryption algorithms, digital signatures
CO4	Analyze the concept of message integrity and the algorithms for checking the integrity of data.
CO5	Understand and analyze the existing cryptosystem used in networking

Note : The students have to perform minimum 10 Practicals based on given syllabus

Subject : Deep Learning **Subject Code : BTCME702.1T**

Load	Credit	Total Marks	Internal Marks	University Marks	Total
03Hrs (Theory)	03	100	30	70	100

Aim :To highlight the features of different technologies involved in Data Science

Prerequisite(s): Artificial Intelligence, Machine Learning

Course Objectives:

1	To introduce basic deep learning algorithms.
2	To understand real world problem and solved by deep learning methods.
3	To identify deep learning techniques suitable for a real world problem.

Course Outcomes: At the end of this course student are able to:

CO1	Understand basic of deep learning algorithms
CO2	Represent feed forward Neural Network
CO3	Evaluate the performance of different deep learning models with respect to the optimization, bias variance trade-off, overfitting and underfitting.
CO4	Apply the convolution networks in context with real world problem solving.
CO5	Apply recurrent neural networks in context with real world problem solving.

UNIT I **(09 Hrs)**

Basic of Deep Learning - History of Deep Learning, McCulloch Pitts Neuron, Thresholding Logic, Perceptrons, Perceptron Learning Algorithm and Convergence, Multilayer Perceptrons (MLPs), Representation Power of MLPs, Sigmoid Neurons, Feed forward Neural Networks.

UNIT II **(07 Hrs)**

Training of feed forward Neural Network - Representation Power of Feed forward Neural Networks, Training of feed forward neural network, Gradient Descent, Gradient Descent (GD), Momentum Based GD, Nesterov Accelerated GD, Stochastic GD, AdaGrad, RMSProp, Adam.

UNIT III **(08 Hrs)**

Optimization Algorithm - Gradient Descent (GD), Momentum Based GD, Nesterov Accelerated GD,

Stochastic GD, AdaGrad, RMSProp, Adam, Activation Function and Initialization Methods: Sigmoid, Tanh, Relu, Xavier and He initialization, Regularization: Bias and variance, Overfitting, Hyper parameters tuning, L1 and L2 regularization, Data Augmentation and early stopping, Parameter sharing and tying.

UNIT IV **(06 Hrs)**

Convolution Neural Network (CNN) - Convolutional operation, Pooling, LeNet, AlexNet, ZF-Net, VGGNet, GoogLeNet, ResNet. Visualizing Convolution Neural Networks, Guided Back propagation.

UNIT V **(06 Hrs)**

Recurrent Neural Network (RNN) - Recurrent Neural Networks, Back propagation through Time (BPTT), Vanishing and Exploding Gradients, Long Short Term Memory (LSTM) Cells, Gated Recurrent Units (GRUs).

Text Books:

1. Sandro Skansi, Introduction to Deep Learning ,Springer.
2. Charu C. , Aggarwal. Neural Networks and Deep Learning: A Textbook. Springer. 2019.
3. Ian Goodfellow , Yoshua Bengio and Aaron Courville. Deep Learning. An MIT Press book. 2016.
4. Dr. S Lovelyn Rose, Dr. L Ashok Kumar, Dr.D Karthika Renuka ,Deep Learning using Python,Willey Publication.

Reference Books:

1. Yegnanarayana, B., Artificial Neural Networks PHI Learning Pvt. Ltd, 2009.
2. A. Ravindran, K. M. Ragsdell , and G. V. Reklaitis, Engineering Optimization: Methods and Applications , John Wiley & Sons, Inc. , 2016.

Subject : Blockchain Technology **Subject Code: BTCME702.2T**

Load	Credit	Total Marks	Internal Marks	University Marks	Total
03Hrs (Theory)	03	100	30	70	100

Aim : To give students the understanding of emerging abstract models for Block chain Technology and to familiarize with the functional/operational aspects of cryptocurrency eco-system.

Prerequisite(s): Cryptography and hashing

Course Objectives:

1	To learn blockchain technology and its related concepts
2	To cover various crypto currencies and usage in block chain
3	To cover security features and its implementation
4	To learn smart contract and applications

Course Outcomes:

At the end of this course Student are able to:

CO1	Describe the basic concepts and technology used for blockchain
CO2	Describe the primitives of the distributed computing and cryptography related to blockchain
CO3	Illustrate the concepts of Bitcoin and their usage.
CO4	Implement Ethereum block chain contract.
CO5	Apply security features in blockchain technologie and Use smart contract in real world applications.

Unit -I **[08 Hrs]**

Introduction: Need for Distributed Record Keeping, Modeling faults and adversaries, Byzantine Generals problem, Consensus algorithms and their scalability problems, Nakamoto's concept with Blockchain based cryptocurrency, Technologies Borrowed in Blockchain – hash pointers, consensus, byzantine fault-tolerant distributed computing, digital cash etc.

Unit –II **[07 Hrs]**

Basic Distributed Computing & Crypto primitives: Atomic Broadcast, Consensus, Byzantine Models of fault tolerance, Hash functions, Puzzle friendly Hash, Collison resistant hash, digital signatures, public key crypto, verifiable random functions, Zero-knowledge systems

Unit -III **[07 Hrs]**

Bitcoin basics: Bitcoin blockchain, Challenges and solutions, proof of work, Proof of stake, alternatives to Bitcoin consensus, Bitcoin scripting language and their use

Unit -IV **[07 Hrs]**

Ethereum basics: Ethereum and Smart Contracts, The Turing Completeness of Smart Contract Languages and verification challenges, Using smart contracts to enforce legal contracts, comparing Bitcoin scripting vs. Ethereum Smart Contracts, Writing smart contracts using Solidity & JavaScript

Unit -V **[07 Hrs]**

Privacy, Security issues in Blockchain: Pseudo-anonymity vs. anonymity, Zcash and Zk-SNARKS for anonymity preservation, attacks on Blockchains: Sybil attacks, selfish mining, 51% attacks advent of algorand; Sharding based consensus algorithms to prevent these attacks

TextBook:

1. Bashir, Mastering Blockchain: Distributed ledger technology, decentralization, and smart contracts explained, 2nd Edition, 2nd Revised edition. Birmingham: Packt Publishing, 2018.

References:

- Narayanan, Bonneau, Felten, Miller and Goldfeder, "Bitcoin and Cryptocurrency Technologies – A Comprehensive Introduction", Princeton University Press.
- Josh Thompson, 'Blockchain: The Blockchain for Beginnings, Guild to Blockchain Technology and Blockchain Programming', Create Space Independent Publishing Platform, 2017.
- Imran Bashir, "Mastering Blockchain: Distributed ledger technology, decentralization, and smart contracts explained", Packt Publishing.
- Merunas Grincalaitis, "Mastering Ethereum: Implement Advanced Blockchain Applications Using Ethereum-supported Tools, Services, and Protocols", Packt Publishing.
- Prof. Sandip Chakraborty, Dr. Praveen Jayachandran, "Blockchain Architecture Design And Use Cases"[MOOC], NPTEL: https://nptel.ac.in/courses/106/105/106105184/

Subject : Augmented & Virtual Reality **Subject Code: BTCME702.3T**

Load	Credit	Total Marks	Internal Marks	University Marks	Total
03Hrs (Theory)	03	100	30	70	100

Aim : To highlight & explore the research approach on Augmented & virtual reality.

Prerequisite(s): Programming, Computer Graphics

Course Objectives:

1	To know basic concepts of virtual reality
2	To understand visual computation in computer graphics
3	To understand interaction between system and computer
4	To know application of VR in Digital Entertainment
5	To know issues & challenges of augmented reality

Course Outcomes:

At the end of this course student are able to:

CO1	Understand the basics of Augmented and Virtual reality systems with the help of input and output devices
CO2	Summarize the basic concepts and hardware & software of Augmented Reality system
CO3	Analyze manipulation, navigation and interaction of elements in the virtual world using tools.
CO4	Apply the concept of technology in the real world applications of entertainment.
CO5	Accept the challenges in applications of augmented and virtual reality.

UNIT I: **(08Hrs)**

Introduction of Virtual Reality: Fundamental Concept and Components of Virtual Reality , Primary Features and Present Development on Virtual Reality, Multiple Models of Input and Output Interface in Virtual Reality: Input - Tracker, Sensor, Digital Glove, Movement Capture, Video-based Input, 3D Menus & 3DScanner, Output -Visual /Auditory / Haptic Devices.

UNIT II: **(07 Hrs)**

Visual Computation in Virtual Reality: Fundamentals of Computer Graphics, Software and Hardware Technology on Stereoscopic Display, Advanced Techniques in CG: Management of Large Scale Environments & Real Time Rendering.

UNIT III: **(07 Hrs)**

Interactive Techniques in Virtual Reality: Body Track, Hand Gesture, 3D Manus, Object Grasp. Development Tools and Frameworks in Virtual Reality: Frameworks of Software Development Tools

in VR. X3D Standard; Vega, MultiGen – Virtools

UNIT IV: **(07 Hrs)**

Application of VR in Digital Entertainment: VR Technology in Film & TV Production,VR Technology in Physical Exercises and Games, Demonstration of Digital Entertainment by VR.

UNIT V: **(07 Hrs)**

Augmented and Mixed Reality: Taxonomy - technology and features of augmented reality, difference between AR and VR, Challenges with AR, AR systems and functionality, Augmented reality methods, visualization techniques for augmented reality.

Textbooks:

- Burdea, G. C., P. Coffet., "Virtual Reality Technology", Second Edition, Wiley-IEEE Press, 2003/2006.
- Alan B. Craig, "Understanding Augmented Reality, Concepts and Applications", Morgan Kaufmann, 2013.

References:

- Alan Craig, William Sherman, Jeffrey Will, "Developing Virtual Reality Applications, Foundations of Effective Design", Morgan Kaufmann, 2009

Subject: Salesforce Technology **Subject Code: BTCME702.4T**

Load	Credit	Total Marks	Internal Marks	University Marks	Total
03Hrs (Theory)	03	100	30	70	100

Aim: To provide a comprehensive understanding of the Salesforce platform, its core features, and its various components.

Prerequisite: OOPS (Object Oriented Programming) or any programming language.

Course Objectives:

1.	To learn how to create and customize objects, fields and records; building workflows and automation processes.
2.	To learn designing and managing reports and dashboards; and utilizing the Salesforce AppExchange.
3.	To provide knowledge and hands-on experience in programming using Apex (Salesforce's proprietary programming language)

Course Outcomes:

At the end of this course students are able to:

CO1.	Develop skills in configuring and managing Salesforce orgs.
CO2.	Understanding Salesforce Data Management
CO3.	Implementing automation, security and debugging data.
CO4.	Acquire programming skills in Apex, Salesforce's programming language.
CO5.	Enable to extend and customize Salesforce to meet specific business requirements.

Unit I **[08 Hrs]**

Salesforce Administration:-

INTRODUCTION TO SALESFORCE:-

Cloud Computing, Services of Cloud computing, Types of Cloud, What is Salesforce?, Salesforce Products, How to create Salesforce developer edition account, Walkthrough Salesforce.com platform.

CONFIGURATION AND CUSTOMIZATION:-

Salesforce: Data types, field types and components.

Apps in Salesforce(Standard Apps,Custom Apps),Steps to Create a SalesforceApp, Salesforce tabs,Types of Tab Visibility.Users & User Licenses.

Salesforce Objects, fields & Field Dependency, Profiles & Roles

Unit II **[07 Hrs]**

Relationships In Salesforce, Validation Rule& formula , Approval process , Flows and Process Builder, Page Layouts, Reports and Dashboards.

WorkFlow: Define Workflow, Workflow Rules, Components of Workflow(action , criteria),How to configure Workflow Rule Criteria?, Setup workflow tasks & Email Alerts & Field Updates, Time dependent workflows.

Unit III **[07 Hrs]**

AUTHORIZATION & SHARING DATA:-

Profiles ,Permission Sets, Org-Wide Defaults , Role Hierarchies, Sharing Rules , Manual Sharing , Record Types

DATA MANAGEMENT :- Import and Export Data, Data Loader.

Unit IV **[07 Hrs]**

INTRODUCTION TO APEX:-Collections (List, Map, Set), DML Operations, SOQL And SOSL Controllers In APEX

Apex Triggers: Overview On Triggers, Trigger Events: Before Triggers, After Triggers, Insert Triggers, Update Triggers, Delete Triggers, Undelete Triggers, Trigger context variables, Recursive Triggers. Governor Limits

[07 Hrs]

Unit V

ASYNCHRONOUS APEX: Future Method, Queueable Apex, Scheduled Apex

Batch APEX: Iterable Class, QueryLocator, GetQueryLocator, Start Method, Execute Method, Finish Mehtod, BatchableContext.

Test Class: StartTest, StopTest, Test Class on Apex class and Triggers

Text /Reference Books:-

1. "Salesforce Platform App Builder Certification Handbook" by Siddhesh Kabe and Muhammad Ehsan Khan (Packt Publishing).
2. "Salesforce CRM: The Definitive Admin Handbook" by Paul Goodey (Packt Publishing).
3. "Force.com Enterprise Architecture" by Andrew Fawcett (Packt Publishing).

4. "Mastering Salesforce CRM Administration" by Rakesh Gupta (Packt Publishing).
5. "Salesforce Essentials for Administrators" by Mohith Shrivastava (Packt Publishing).
6. "Learning Salesforce Lightning Application Development" by Mohith Shrivastava (Packt Publishing).
7. "Apex Design Patterns" by Jitendra Zaa (Packt Publishing).
8. "Mastering Apex Programming" by ChamilMadusanka (Packt Publishing).

Subject : Compiler Design **Subject Code: BTCME703.1T**

Load	Credit	Total Marks	Internal Marks	University Marks	Total
03Hrs (Theory)	03	100	30	70	100

Aim: To understand the principles and concepts of Compiler Design.

Prerequisite(s): Students should have the knowledge of computers and mathematics.

Course Objectives:

1	Define the different phases of the Compiler and utilities of Automata.
2	Understand the concept of syntax and semantic Analysis.
3	Understand the target machine's run time environment, its instruction set for code generation and techniques used for code optimization.
4	Understand the Architecture of computer and its use in designing a compiler.

Course Outcomes:

At the end of this course students are able to:

CO1	Define the compiler along with the phases and basic program in LEX
CO2	Understand the parser and its types i.e. Top-Down and Bottom-up parsers and construction of parsing table.
CO3	Implement program based on concept of type checking, parameter passing and Overloading.
CO4	Implement the concept of Code Optimizations and Code Generations.
CO5	Understand the concepts of Object Oriented in Compilers.

Unit I: **[07 Hours]**

Introduction: Phases of compilation and over view, Lexical Analysis(Scanner),Regular languages, finite automata, regular expression, relating regular expression and finite automata, scanner generator (lex, flex)

Unit II **[07 Hours]**

Syntax Analysis (Parser): push-down automata, and top-down parsing, bottom up parsing, ambiguity and parser generator (yacc, bison)

Unit III: **[08 Hours]**

Semantic Analysis: Attribute Grammar, syntax directed definition, evaluation, and flow of attribute in a syntax tree, Symbol Table: Basic Structure, symbol table attributes and managements, Runtime Environment: Procedure activation, parameter passing, value return, memory allocation

Unit IV: **[07 Hours]**

Intermediate Code Generation: Translation of different language features, different types of intermediate forms, code improvement (optimization), control flow, data dependence etc, local optimization, global optimization, peephole optimization etc.

Unit V: **[07 Hours]**

Architecture dependent code improvement: instruction scheduling (for pipelining), loop optimization (for cache memory), Register allocation and target code generation.

Textbooks:

1. Compilers: Principles, Techniques and Tools, V. Aho,R. Sethi, J.Ullman.
2. Lex and Yaac, Levine R. John, Tony Mason and Doug Brown.

Subject : Natural Language Processing **Subject Code : BTCME703.2T**

Load	Credit	Total Marks	Internal Marks	University Marks	Total
03Hrs (Theory)	03	100	30	70	100

Aim : To understand the implementation of various Natural Language Processing

Prerequisite(s): TOC, Compiler Design, programming language

Course Objectives:

1	To introduce the basic concepts and applications of Natural Language Processing (NLP)
2	To provide an understanding of the challenges in NLP and their solutions
3	To teach the different techniques and algorithms used in NLP, such as text classification, information retrieval and extraction, syntactic and semantic analysis, and deep learning models
4	To enable students to analyze text data and build NLP models
5	To equip students with the skills to evaluate and compare different NLP techniques and algorithms

Course Outcome:

At the end of this course Student are able to:

CO1	Understand the basic concepts and applications of Natural Language Processing (NLP)
CO2	Identify the challenges in NLP and evaluate the solutions to these challenges
CO3	Analyze and preprocess text data for NLP tasks
CO4	Apply different NLP techniques and algorithms such as text classification, information retrieval and extraction, syntactic and semantic analysis, and deep learning models
CO5	Evaluate and compare different NLP techniques and algorithms using appropriate metrics

UNIT I: **(08Hrs)**

Introduction to NLP: Definition and scope of NLP, Historical overview and applications of NLP, Challenges in NLP and their solutions, Basic concepts in linguistics and language processing, Text preprocessing and normalization

UNIT II: **(07 Hrs)**

Language Models and Text Classification: Language modeling and n-gram models, Classification and categorization of text data, Text classification algorithms such as Naive Bayes, Decision Trees, and Support Vector Machines (SVM), Evaluation measures for text classification.

UNIT III: **(07 Hrs)**

Information Retrieval and Extraction: Information retrieval models such as vector space model and probabilistic model, Retrieval of relevant documents using query expansion, Named Entity Recognition (NER), Relation Extraction and Open Information Extraction (OIE)

UNIT IV: **(07 Hrs)**

Syntactic and Semantic Analysis: Parts of Speech (POS) tagging and parsing, Dependency Parsing, Semantic Analysis and Sentiment Analysis, Word Embeddings and Semantic Similarity

UNIT V: **(07 Hrs)**

Advanced Topics in NLP: Neural Network models for NLP tasks , Deep Learning models for NLP tasks, Natural Language Generation (NLG), Dialogue Systems and Chatbots

Textbooks:

- Speech and Language Processing" by Daniel Jurafsky and James H. Martin
- Natural Language Processing" by Jacob Eisenstein

References:

- "Foundations of Statistical Natural Language Processing" by Christopher D. Manning and Hinrich Schütze
- "Natural Language Processing with Python" by Steven Bird, Ewan Klein, and Edward Loper

Subject: Introduction to Software Testing Subject Code: BTCME703.3T

Load	Credit	Total Marks	Internal Marks	University Marks	Total
03Hrs (Theory)	03	100	30	70	100

Aim : Nil

Prerequisite(s): Software Engineering

Course Objectives:

1	To understand the software testing principles and software quality
2	To understand various software testing methods
3	To develop the software which adheres to the standard benchmarks
4	To develop the skills by utilizing various testing tools and solve the different issues

Course Outcomes:

At the end of this course student are able to:

CO1	Understand the role of software testing and quality assurance
CO2	Acquire the knowledge of test case strategies using white and black box approach
CO3	Understand and apply the different levels of testing
CO4	Develop & analyze the software testing process & management skills
CO5	Develop & Evaluate the software testing tools for commercial applications.

UNIT I: **(08Hrs)**

Fundamentals of Testing and its current state of art: Testing as an Engineering Activity -Role of Process in Software Quality Testing as a Process Basic Definitions, Software Testing Principles The Tester's Role in a Software Development Organization, Origins of Defects Developer/Tester Support for Developing a Defect Repository.

UNIT II: **(07 Hrs)**

Test Case Design Strategies, Using Black Box Approach to Test Case Design Random Testing, Equivalence Class partitioning state-based testing, cause effect graphing, error guessing, compatibility testing, user documentation testing, and domain testing using white-box approach to test design

UNIT III: **(07 Hrs)**

The Need for Levels of Testing, Unit Test, The Test Harness, Integration tests, Scenario testing, defect

bash elimination, system testing, types of system testing, testing systems, usability and accessibility testing

UNIT IV: **(07 Hrs)**

Test Management Test Planning, Test Plan Components Test Plan Attachments Locating Test Items, test management, test process, reporting test results, Introducing the test specialist, skills needed by a test specialist, building a testing group.

UNIT V: **(07 Hrs)**

Controlling and Monitoring Software test automation, requirements for a test tool, challenges in automation, Test metrics and measurements, project, progress and productivity metrics, Status Meetings, Reports and Control Issues, Criteria for Test Completion, SCM, evaluating software quality.

Textbooks:

1. Boris Beizer, "Software Testing Techniques", Second Edition, Dreamtech, 2003.
2. Elfriede Dustin, "Effective Software Testing", First Edition, Pearson Education, 2003.

References:

1. Aditya P Mathur, "Foundations of Software Testing", Second Edition, Pearson Education, 2008.
2. Renu Rajani, Pradeep Oak, "Software Testing Effective Methods, Tools and Techniques", Tata McGraw Hill, 2004.

Subject: The Joy of Computing using Python Subject Code: BTCME704.1T
(OPEN ELECTIVE –II)

Load	Credit	Total Marks	Internal Marks	University Marks	Total
03Hrs (Theory)	03	100	30	70	100

Aim : To understand the implementation of various The Joy of Computing using Python

Prerequisite(s): C Language

Course Objective/Learning Objective:

1	Develop a foundational understanding of Python programming language and its applications in data analysis and machine learning.
2	Gain knowledge of data structures and libraries in Python to manage data and perform advanced data analysis tasks.
3	Develop an understanding of algorithm design and analysis and apply it to solve problems.
4	Apply problem-solving strategies and techniques using Python to solve real-world problems
5	Gain knowledge of advanced Python topics and their applications in various domains

Course Outcome:
At the end of this course Student are able to:

CO1	Develop proficiency in Python programming language and apply it to solve computational problems.
CO2	Use data structures and libraries in Python to manage data and perform advanced data analysis tasks.
CO3	Design and analyze algorithms to solve problems efficiently and effectively.
CO4	Apply problem-solving strategies and techniques using Python to solve real-world problems.
CO5	Demonstrate knowledge of advanced Python topics and their applications in various domains.

UNIT I: **(08Hrs)**

The Joy of Computing using Python: Introduction to Python Programming, Basics of Python programming language, Variables and data types, operators, and expressions, Control structures (if, for, while) and functions, Python Data Structures and Libraries

UNIT II: **(07 Hrs)**

Lists, tuples, sets, and dictionaries: File handling and libraries (NumPy, pandas, etc.), Introduction to object-oriented programming in Python, Algorithm Design and Analysis.

UNIT III: **(07 Hrs)**

Introduction to algorithms and algorithm analysis: Searching and sorting algorithms, Recursion and dynamic programming, Problem-Solving with Python.

UNIT IV: **(07 Hrs)**

Developing problem-solving strategies and techniques: Developing programs for real-world problems, Debugging, testing, and code documentation, Advanced Python Topics

UNIT V: **(07 Hrs)**

Regular expressions and string manipulation: Networking programming with Python, Web scraping and data mining, Introduction to machine learning with Python

Textbooks:

- Downey, A. (2015). Think Python: How to Think Like a Computer Scientist, 2nd ed., O'Reilly Media.
- Severance, C. (2016). Python for Everybody: Exploring Data in Python 3, 1st ed., CreateSpace Independent Publishing Platform.

References:

- Zelle, J. M. (2016). Python Programming: An Introduction to Computer Science, 3rd ed., Franklin, Beedle & Associates Inc.
- Van Rossum, G., & Drake, F. L. (2010). The Python Language Reference Manual, CreateSpace Independent Publishing Platform.
- McKinney, W. (2017). Python for Data Analysis: Data Wrangling with Pandas, NumPy, and IPython, 2nd ed., O'Reilly Media.

Subject: Database Management System Subject Code : BTCME704.2T
(OPEN ELECTIVE – II)

Load	Credit	Total Marks	Internal Marks	University Marks	Total
03 Hrs.(Theory)	03	100	30	70	100

Aim: To understand basic concepts of Database Management System.

Prerequisite(s): NIL

Course Objective/Learning Objective:

1.	To introduce a general idea of a database management system.
2.	To develop skills to implement real life applications that involve database handling.
3.	To provide opportunities in subject areas of data handling and managing techniques

Course Outcome:
At the end of this course Student are able to:

CO1.	Understand the basics of DBMS to analyze an information problem in the form of an Entity relation diagram and design an appropriate data model for it.
CO2.	Demonstrate basics of File organizations and its types
CO3.	Interpret functional dependencies and various normalization forms
CO4.	Perform basic transaction processing and management
CO5	Demonstrate SQL queries to perform CRUD (Create, Retrieve, Update, Delete) operations on database

UNIT I: (08Hrs)

Introduction to DBMS - Purpose of Database Systems, Database systems Applications, view of data,

Database Languages, Database system structure, data methods, Database Design,& ER Model : Entity, Attributes, Relationships, Constraints, Keys, Design Process, ER Models, E-R Diagram.

UNIT II: (07 Hrs)

File organizations and its types, indexing, types of indexing, hashing, hashing techniques.

UNIT III: (07 Hrs)

Functional Dependency (FD) – data integrity rules, functional dependency, need of normalization, first normal form, second normal form, third normal form

UNIT IV: (07 Hrs)

Database Transaction Processing : transaction system concepts, desirable properties(ACID) of transactions, schedules, serializability of schedules, concurrency control, recoverability and Deadlock handling.

UNIT V: (07 Hrs)

SQL Concepts - Basics of SQL, DDL, DML, DCL, structure – creation, alteration, defining constraints, Functions - aggregate functions, Built-in functions –numeric, date, string functions, set operations, Use of group by, having, order by, join and its types, Exist, Any, All

Textbooks:

1. Abraham Silberschatz, Henry F. Korth and S. Sudarshan, Database System Concepts 4th Ed, McGraw Hill, 2010
2. Ramez Elmasri and Shamkant B. Navathe, Fundamentals of Database Systems (5/e), Pearson Education, 2008
3. Raghu Ramakrishnan and Johannes Gehrke, Database Management Systems (3/e), McGraw Hill

References:. .

- Peter Rob and Carlos Coronel, Database Systems- Design, Implementation and Management (7/e), Cengage Learning, 2007.

Subject : Data Visualization Subject Code:BTCME704.3T
(OPEN ELECTIVE- II)

Load	Credit	Total Marks	Internal Marks	University Marks	Total
03Hrs (Theory)	03	100	30	70	100

Aim : To highlight the features of different technologies involved in data visualization.

Prerequisite(s): Mathematics, Graphs

Course Objectives:

1	To learn different statistical methods for Data visualization.
2	To learn basics of R and Python.
3	To learn usage of Watson studio.
4	To learn about packages Numpy, pandas and matplotlib.
5	To learn functionalities and usages of Seaborn.

Course Outcomes:
At the end of this course student are able to:

CO1	Apply statistical methods for Data visualization.
CO2	Gain knowledge on R and Python
CO3	Understand usage of various packages in R and Python.
CO4	Demonstrate knowledge of Watson studio.
CO5	Apply data visualization tools on various data sets.

UNIT I: **(10Hrs)**

Introduction to Statistics : Introduction to Statistics, Difference between inferential statistics and descriptive statistics, Inferential Statistics- Drawing Inferences from Data, Random Variables, Normal Probability Distribution, Sampling, Sample Statistics and Sampling Distributions. R overview and Installation- Overview and About R, R and R studio Installation, Descriptive Data analysis using R, Description of basic functions used to describe data in R.

UNIT II: **(07 Hrs)**

Data manipulation with R: Data manipulation packages, Data visualization with R. Data visualization in Watson Studio: Adding data to data refinery, Visualization of Data on

Watson Studio.

UNIT III: **(05 Hrs)**

Python: Introduction to Python, How to Install, Introduction to Jupyter Notebook, Python scripting basics, Numpy and Pandas.

UNIT IV: **(08 Hrs)**

Data Visualization Tools in Python- Introduction to Matplotlib, Basic plots using matplotlib, Specialized Visualization Tools using Matplotlib, Advanced Visualization Tools using Matplotlib- Waffle Charts, Word Clouds

UNIT V: **(06 Hrs)**

Introduction to Seaborn: Seaborn functionalities and usage, Spatial Visualizations and Analysis in Python with Folium, Case Study

Textbooks:

1. Core Python Programming - Second Edition, R. Nageswara Rao, Dreamtech Press.
2. R Graphics Essentials for Great Data Visualization by Alboukadel Kassambara

References:

1. Phuong Vo.T.H, Martin Czygan, Ashish Kumar, Kirthi Raman, Python Data Analytics and Visualization. A course in three modules, Packt Publishing 2017.

Subject : Project Work/Industry Project (Phase I) **Subject Code: BTCME705P**

Load	Credit	Total Marks	Internal Marks	University Marks	Total
06 Hrs (Practical)	03	100	50	50	100

Subject : Report Writing Activity (Audit Course) **Subject Code: BTCME706P**

Load	Credit	Total Marks	Internal Marks	University Marks	Total
02 Hrs (Practical)	-	-	-	-	-

***The students may write the report using any open source tools**