BHARATHIDASAN UNIVERSITY TIRUCHIRAPPALLI -620 024.

MASTER OF COMPUTER APPLICATION
CHOICE BASED CREDIT SYSTEM -
LEARNING OUTCOMES BASED CURRICULUM FRAMEWORK (CBCS - LOCF)

(Applicable to the candidates admitted from the calendar year 2024 onwards)

Sem.	Course	Course Title	Ins. Hrs	Credit	Exam Hrs	Marks		Total
						Int.	Ext	
I	Core Course I (CC)	Mathematical Foundations of Computer Applications	6	5	3	25	75	100
	Core Course II (CC)	Design and Analysis of Algorithms	6	5	3	25	75	100
	Core Course III (CC)	Open Source Programming	6	5	3	25	75	100
	Core Practical I (CP)	Algorithm Lab (Choice for Tools)	3	2	3	40	60	100
	Core Practical II (CP)	Open Source Programming	3	2	3	40	60	100
	Elective Course I (EC)	OOAD and Design Patterns	6	4	3	25	75	100
	Total		**30**	**23**	**-**	**-**	**-**	**600**

MANDATORY BRIDGE COURSE FOR STUDENTS OF NON-COMPUTER SCIENCE STREAM – 1st SEMESTER

Sem.	Bridge Courses	Bridge Courses Title	Ins. Hrs	Credit	Exam Hrs	Marks		Total
						Int.	Ext	
I	Bridge Course I (BC)	Programming in C and C++		4	3	25	75	100
	Bridge Course II (BC)	Fundamental of Data Structures		4	3	25	75	100
	Bridge Practical I (BP)	Software Lab: Programming in C and C++		2	3	40	60	100
		Total		**10**	**-**	**-**	**-**	**300**

2nd SEMESTER

Sem.	Course	Course Title	Ins. Hrs	Credit	Exam Hrs	Marks		Total
						Int.	Ext	
II	Core Course IV (CC)	Emerging Technologies in Data Processing	6	5	3	25	75	100
	Core Course V (CC)	Advanced Operating Systems	5	5	3	25	75	100
	Core Course VI (CC)	Computer Graphics and Animation	5	5	3	25	75	100
	Core Practical III (CP)	Operating System	3	2	3	40	60	100
	Core Practical IV (CP)	Computer Graphics and Animation	3	2	3	40	60	100
	Elective Course II (EC)	Internet of Things	5	4	3	25	75	100
		Total	**27**	**23**	**-**	**-**	**-**	**600**

MANDATORY BRIDGE COURSE FOR STUDENTS OF NON-COMPUTER SCIENCE STREAM – 2nd SEMESTER

Sem.	Bridge Courses	Bridge Courses Title	Ins. Hrs	Credit	Exam Hrs	Marks		Total
						Int.	Ext	
II	Bridge Course III (BC)	Web Design		4	3	25	75	100
	Bridge Course IV (BC)	Digital Electronics		4	3	25	75	100
	Bridge Practical II (BP)	Web Design		2	3	40	60	100
		Total		**10**	**-**	**-**	**-**	**300**

3rd SEMESTER

Sem.	Course	Course Title	Ins. Hrs	Credit	Exam Hrs	Marks		Total
						Int.	Ext	
III	Core Course VII (CC)	Compiler Design	5	5	3	25	75	100
	Core Course VIII (CC)	Machine Learning Techniques	6	5	3	25	75	100
	Core Course IX (CC)	Big data Analytics	5	5	3	25	75	100
	Core Practical V (CP)	Machine Learning Techniques	3	2	2	40	60	100
	Core Practical VI (CP)	Big data Analytics	3	2	2	40	60	100
	Elective Course III (EC)	Natural Language Processing	5	4	3	25	75	100
		Total	**27**	**23**	**-**	**-**	**-**	**600**

MANDATORY BRIDGE COURSE FOR STUDENTS OF NON-COMPUTER SCIENCE STREAM – 3rdSEMESTER

Sem.	Bridge Courses	Bridge Courses Title	Ins. Hrs.	Credit	Exam Hrs.	Marks		Total
						Int.	Ext.	
III	Bridge Course – V (BC)	Python Programming		4	3	25	75	100
	Bridge Course – VI (BC)	Internet Programming		4	3	25	75	100
	Bridge Practical–III (BP)	Python Programming		2	3	40	60	100
		Total		**10**	-	-	-	**300**

4th Semester

Sem.	Course	Course Title	Ins. Hrs	Credit	Exam Hrs	Marks		Total
						Int.	Ext	
IV	Core Course – X (CC)	Cloud Computing Fundamentals	6	5	3	25	75	100
	Core Course XI (CC)	Managerial Skills	6	5	3	25	75	100
	Project Work	Dissertation	12	11	-	20	80	100
	Total		**24**	**21**	-	-	-	**300**
Grand Total				**90/ 120#**				**2100/ 3000#**

PROGRAMME OBJECTIVES:

.

- To Exhibit professionalism, ethical attitude, communication skills, team work in their profession and adapt to current trends by engaging in lifelong learning
- To continue a lifelong professional development in computing that contributes in self and societal growth
- To Produce knowledgeable and skilled human resources which are employable in IT and ITES.

PROGRAMME OUTCOMES:

After the successful completion of M.C.A. Programme, the Graduates will be able to

- Develop software solutions to problems across a broad range of application domains through analysis and design.
- Identify, formulate, and solve complex computing problems reaching substantiated conclusions using fundamental principles of mathematics, computing sciences.
- Design and evaluate solutions for complex computing problems, design and evaluate systems, components, or processes that meet specified needs with societal focus
- Able to understand of professional, ethical, legal, security and social issues and responsibilities.
- Use research-based knowledge and methods to conduct investigations on complex problems and provide valid conclusions.
- Identify a timely opportunity to use innovation to pursue and create value and wealth for the betterment of the Society and Nation.

First Year **CORE COURSE I** **Semester I**

MATHEMATICAL FOUNDATIONS OF COMPUTER APPLICATIONS

Code: **(Theory)** **Credit: 5**

COURSE OBJECTIVES:

- Understand the Mathematical logics and Predicate Calculus
- To learn the mathematical foundations applicable to computing
- Understand the basics of language and its Grammar

UNIT – I MATHEMATICAL LOGIC:

Statements and notation – Connectives – Negation – Conjunction – Disjunction – Statement formulae and truth tables – Conditional and Biconditional - Well formed formulas – Tautologies – Equivalences of formula – Duality Law. Predicate Calculus: Predicates – Statement functions – variables – Quantifiers – predate formulae – free & bound variables.

UNIT – II BASIC CONCEPTS OF SET THEORY:

Notation – Inclusion of equality of sets – power set – operation on sets – Venn diagrams – Cartesian products. Relations and Ordering: Relations – Properties of Binary relation in a set – Relation matrix and graph – Equivalence relations – Composition of binary relations – Partial Ordering.

UNIT – III GROUPS:

Definition and examples – Sub groups – Homomorphism – Cosets – Normal Subgroups.

UNIT – IV GRAPH THEORY:

Basic Definitions – Paths, Reachability Connectedness – Matrix Representation of graphs – Trees.

UNIT – V GRAMMARS AND LANGUAGES:

Introduction – alphabet, words, languages – regular expressions, regular languages - Finite state Automata – Grammars – Gödel Numbers.

UNIT – VI CURRENT CONTOURS (For continuous internal assessment only):

Contemporary Developments Related to the Course during the Semester Concerned.

REFERENCES:

1. Lipschuta Seymour, Lipson Marc, *Discrete Mathematics*, Schaum's outline Series, Tata McGraw Hill, New Delhi, 2007,
2. Trembley, Manohar, *Discrete Mathematics Structures with Applications to Computer Science*, Tata McGraw Hill, New Delhi, 1997
3. Kolman Bernard, Robert C.Busby, *Discrete Mathematical Structures for Computer Science*, Second EditionPHI, 2014
4. Hopcroft, Joseph E. Ullman, Jeffery D, *Introduction to Automata Theory Languages and Computations*, Narosa Publishing House, New Delhi, 2014
5. Levin Oscar, *Discrete Mathematics An Open Introduction*, Third Edition, 2013
6. E. Lehman, F. T. Leighton, and A. R. Meyer, Mathematics for Computer Science,2013.(WebLink:http://courses.csail.mit.edu/6.042/spring13/mcs.pdf)
7. R. L. Graham, D. E. Knuth, and O. Patashnik, Concrete Mathematics, Pearson,1994. (Web Link : www.maths.ed.ac.uk/~aar/papers/knuthore.pdf)
8. Aho and J. Ullman, Foundations of Computer Science, W. H. Freeman, 1992. (Web Link : http://infolab.stanford.edu/~ullman/focs.html)
9. http://nptel.ac.in/courses.php?disciplineId=111
10. http://www.class-central.com/subject/math(MOOCs)
11. http://ocw.mit.edu/courses/mathematics/

COURSE OUTCOMES:

At the end of the course, the students will be able to

- Solve the problems using truth table technique, rules of inference method.
- Apply the concepts of Set theory and Relation
- Demonstrate the basics of groups and sub groups.
- Apply the Graph theory concepts in Computer Network and Computer Graphics.
- Ability to understand and construct languages.

*****

First Year | **CORE COURSE II** | **Semester I**

DESIGN AND ANALYSIS OF ALGORITHMS

Code: | **(Theory)** | **Credit: 5**

COURSE OBJECTIVES:

- To analyse the performance of algorithms under various scenarios.
- To learn mathematical background for algorithm analysis & solving the recurrence equations.
- To learn various algorithm design techniques.

UNIT – 1 INTRODUCTION:

Fundamentals of Algorithmic Problem Solving - Time Complexity - Space complexity with examples - Growth of Functions -Asymptotic Notations: Need, Types - Big Oh, Little Oh, Omega, Theta - Properties - Complexity Analysis Examples -Performance measurement - Instance Size, Test Data, Experimental setup.

UNIT – II MATHEMATICAL FOUNDATIONS:

Solving Recurrence Equations - Substitution Method - Recursion Tree Method - Master Method - Best Case - Worst Case -Average Case Analysis - Sorting in Linear Time - Lower bounds for Sorting: - Counting Sort - Radix Sort - Bucket Sort

UNIT – III BRUTE FORCE AND DIVIDE-AND-CONQUER:

Brute Force: Travelling Salesman Problem - Knapsack Problem - Assignment Problem - Closest Pair and Convex Hull Problems - Divide and Conquer Approach:- Binary Search - Quick Sort - Merge Sort - Strassen's Matrix Multiplication.

UNIT – IV GREEDY APPROACH AND DYNAMIC PROGRAMMIN:

Greedy Approach: Optimal Merge Patterns- Huffman Code - Job Sequencing problem- -- Tree Vertex Splitting Dynamic Programming:– Dice Throw-- Optimal Binary Search Algorithms.

UNIT – V BACKTRACKING AND BRANCH AND BOUND:

Backtracking:- 8 Queens - Hamiltonian Circuit Problem - Branch and Bound - Assignment Problem - Knapsack Problem:- Travelling Salesman Problem - NP Complete Problems - Clique Problem - Vertex Cover Problem .

REFERENCES:

1. Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest and Clifford Stein, "Introduction to Algorithms", Third
2. Edition, PHI Learning Private Limited, 2012.
3. Alfred V. Aho, John E. Hopcroft and Jeffrey D. Ullman, "Data Structures and Algorithms David E. Goldberg, "Genetic Algorithm In Search Optimization And Machine Learning" Pearson Education India, 2013.
4. AnanyLevitin, "Introduction to the Design and Analysis of Algorithms", Third Edition, Pearson Education, 2012.
5. Ellis Horowitz, SartajSahni and SanguthevarRajasekaran, Fundamentals of Computer Algorithms, Second Edition, Universities Press, 2007

COURSE OUTCOMES:

On completion of the course the student will be able to

- Determine the suitable algorithmic design technique for a given problem.
- Identify the limitations of algorithms in problem solving
- Analyze the efficiency of the algorithm based on time and space complexity.
- Implement asymptotic notations to analyze worst-case and average case running times of algorithms.
- Interpret the fundamental needs of algorithms in problem solving.

*****

First Year **CORE COURSE III** **Semester I**

1) OPEN SOURCE PROGRAMMING

Code: **(Theory)** **Credit: 5**

COURSE OBJECTIVES:

- To understand the basics of open source software
- To create dynamic web applications using PHP, MySQL
- To create web applications based on PHP and AJAX

UNIT – I OPEN SOURCE & FREE SOFTWARE LICENSING:

Open Source Licensing: Basic Principles of Copyright Law – Contract and Copyright – Open Source Software Licensing – Issues with Copyrights and Patents – Open Source Definition – MIT License – BSD License – Apache License – GNU General Public License –

Free and Open Source Software Development: Models of Open Source and Free Software Development – Choosing an Open Source or Free Software License

UNIT – II BASICS OF PHP PROGRAMMING:

Basics of PHP Programming: Introduction – syntax and variables – controls and functions – passing information between pages – strings – numbers – arrays, array functions and advanced array functions

UNIT – III ADVANCED FEATURES AND TECHNIQUES:

Advanced PHP Programming: Object-Oriented Programming with PHP – String and Regular Expression Functions – Filesystem and System Functions – Sessions, Cookies and HTTP – Exceptions and Error Handling

UNIT – IV PHP AND MySQL:

Why PHP and MySQL? – Server-Side Web Scripting – SQL Tutorial – MySQL Database Administration – PHP/MySQL Functions – Displaying Queries in Tables – Building Forms from Queries

UNIT – V PHP &AJAX AND GITHUB HOSTING SERVICE:

PHP and AJAX: JavaScript and AJAX Client – JavaScript and DOM – XML Http Request Object – AJAX form validation – Uploading a file using AJAX – Displaying a table in AJAX – Building Pagination using PHP and AJAX

Hosting Open Source Projects using Github: Introduction – Viewing Github Graphs- Editing Files – Collaborating on Pull Requests – Creating a Repository – Configuring a Repository

UNIT – VI CURRENT CONTOURS (For continuous internal assessment only):

Contemporary Developments Related to the Course during the Semester Concerned.

REFERENCES:

1. Andrew M. St. Laurent, 'Understanding Open Source & Free Software Licensing', O'Reilly Media, 2004.
2. Tim Converse and Joyce Park, 'PHP 5 and MySQL Bible', Wiley Publishing, 2004.
3. K.Meena, R.Slvakumar,A .B.KarthickAnand Babu, Web Programming with PHP and Mysql, Himalaya Publications. Mumbai, 2012.(ISBN :978- 93 - 5051 - 581• 5)
4. BogdanBrinzarea-Lamandi, CristianDarie and Audra Hendrix, 'AJAX and PHP', Packt Publishing, 2009.
5. Peter Bell and Brent Beer, 'Introducing Github: a Non-Technical Guide', O'Reilly Media, 2014
6. Gordon Haff, 'How Open Source Ate Software', Apress, 2018.
7. Rao M. N., 'Fundamentals of Open Source Software', PHI Learning Pvt Ltd, 2014.
8. Robin Nixon, 'Learning PHP, MySQL & JavaScript with jQuery, CSS & HTML5', O'Reilly Media, 2015.
9. Steven Holzner, 'PHP: The Complete Reference', McGraw Hill Education, 2017.
10. https://swayam.gov.in/nd2_aic20_sp32/
    https://www.tutorialspoint.com/php/php_and_mysql.htm
11. https://docs.github.com/en/get-started/quickstart/hello-world
12. https://developer.mozilla.org/en-US/docs/Web/Guide/AJAX/Getting_Started

COURSE OUTCOMES:

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

- Understand the significance of open-source principles and practices
- Understand the fundamentals of PHP
- Develop object oriented based applications using PHP
- Develop web applications using PHP, MySQL and AJAX
- Host open-source projects using Github

*****

First Year	**CORE PRACTICAL I ALGORITHMS**	**Semester I**
Code:	**(Practical)**	**Credit: 2**

IMPLEMENT THE FOLLOWING USING C/C++/JAVA

1. Sort a given set of elements using the quick sort method and determine the time required to sort the elements
2. Implement merge sort algorithm to sort a given set of elements and determine the time required to sort the elements
3. Implement 0/1 Knapsack problem using Dynamic Programming
4. From a given vertex in a weighted connected graph, find shortest paths to other vertices using Dijkstra's algorithm.
5. Perform various tree traversal algorithms
6. Find Minimum Cost Spanning Tree of a given undirected graph using Prim's algorithm.
7. Implement N Queen's problem using Back Tracking.

*****

First Year **CORE PRACTICAL II** **Semester I**

OPEN SOURCE PROGRAMMING

Code: **(Practical)** **Credit: 2**

DEVELOP AND IMPLEMENT THE FOLLOWING PROGRAMS:

1. Develop a server side PHP program that displays marks, total, grade of a student in tabular format by accepting user inputs for name, number and marks from a HTML form.
2. Develop a PHP program that adds products that are selected from a web page to a shopping cart.
3. Develop a PHP program to access the data stored in a mysql table.
4. Develop a PHP program interface to create a database and to insert a table into it.
5. Develop a PHP program using classes to create a table.
6. Develop a PHP program to upload a file to the server.
7. Develop a PHP program to create a directory, and to read contents from the directory.
8. Develop a shell program to find the details of an user session.
9. Develop a shell program to change the extension of a given file.
10. Create a mysql table and execute queries to read, add, remove and modify a record from that table.

*****

First Year **ELECTIVE COURSE I** **Semester I**

OOAD AND DESIGN PATTERNS

Code: **(Theory)** **Credit: 4**

COURSE OBJECTIVE:

- To describe the object-oriented software development process, including object- oriented methodologies and work flow
- To emphasizes on Object Oriented software design and application of design patterns
- To explain various UML diagrams

UNIT – I:

INTRODUCTION TO UML: Introduction to object oriented concepts like inheritance, Polymorphism, Information hiding, Importance of modelling, Principles of modelling, Object oriented modelling, An overview of UML, Conceptual model of the UML, Architecture, Software development life cycle. **BASIC STRUCTURAL MODELING:** Classes: Terms and concepts, Common modelling techniques; Relationships Modelling simple dependencies, Single inheritance and structural relationships; Common mechanisms and diagrams.

ADVANCED STRUCTURAL MODELING: Advance classes, Advance relationships, Interfaces, Types and Roles, Packages, Instances.

UNIT – II:

THE OBJECT-ORIENTED DESIGN PROCESS: The object and class Concepts, Identifying classes, Identifying responsibilities, Relationships between Classes, Use Cases, CRC cards, UML class diagrams, Sequence diagrams, State diagrams, Using Java doc for design documentation, Case Study: A voice mail system.

UNIT – III:

GUIDELINES FOR CLASS DESIGN: An overview of the date classes in the java library, designing a day class, the importance of encapsulation, analyzing the quality of an interface, programming by contract, unit testing.
INTERFACE TYPES AND POLYMORPHISM: The icon interface type, polymorphism, drawing shapes, the comparable interface type, the comparator interface type, anonymous classes, frames and user interface components, user interface actions, timers, designing an interface type.

UNIT – IV:

PATTERNS AND GUI PROGRAMMING: Iterators, the pattern concept, the observer pattern, layout managers and the strategy pattern, components, containers and the composite pattern, scroll bars and the decorator pattern, how to recognize patterns, putting patterns to work. **INHERITANCE AND ABSTRACT CLASSES:** The concept of inheritance, graphics programming with inheritance, abstract classes, the template method pattern, protected interfaces, the hierarchy

of swing components, the hierarchy of standard geometric shapes, the hierarchy of exception classes, when not to use inheritance.

UNIT – V:

FRAMEWORKS: Frameworks, applets as a simple framework, the collections framework, a graph editor framework, enhancing the graph editor framework.
MULTITHREADING: Thread basics, Thread synchronization, Animations.
MORE DESIGN PATTERNS: The Adapter pattern, Actions and the command pattern, the factory method pattern, the proxy pattern, the singleton pattern, the visitor pattern, other design patterns.

UNIT – VI CURRENT CONTOURS (For continuous internal assessment only):

Contemporary Developments Related to the Course during the Semester Concerned

REFERENCES:

1. Grady Booch, James Rumbaugh, Ivar Jacobson (2009), The Unified Modeling Language Userguide, 2nd edition, Pearson Education, New Delhi, India.
2. Cay Horstmann (2004), Object-Oriented Design and Patterns, Wiley India edition, New Delhi,India.
3. Meilir Page-Jones (2000), Fundamentals of Object Oriented Design in UML, Pearson Education and NewYork.
4. Craig Larman (2005), An introduction to Object –Oriented Analysis and Design and Unified Process Appling UML and Patterns, 3rdedition, Pearson Education, New Delhi, India.
5. John W. Satzinger, Robert B Jackson, Stephen D Burd (2004), Object-Oriented Analysis and Design with the Unified Process, Cengage learning, India.
6. https://www.javatpoint.com/uml-class-diagram
7. https://developer.ibm.com/articles/an-introduction-to-uml/

COURSE OUTCOMES:

At the end of the course student will be able to:

- Analyze the requirements and generate use cases
- Perform Object oriented analysis
- Perform overall design using various UML diagrams
- Understand the Guidelines for Class Design
- Understand different forms of Patterns.

*****

First Year	BRIDGE COURSE I	Semester I
	PROGRAMMING IN C AND C++	
Code:	(Theory)	Credit: 4

COURSE OBJECTIVES:

To provide knowledge of basics elements of C and C++ languages, their specifications, functions, passing of parameters, control constructs; Special features of C language such as Structures and Unions, Pointers and types of files and processing them; Classes and objects in C++ language and the features of C++ and to use them effectives to solve problems.

UNIT – I:

History, Execution of C Program, Constants, Variables and Keywords, Data types, Expressions, constants, variables, Operators, Formatted Console I/O Functions, Conversion Specifications, assignment statements, conditional statements, Looping Statements

UNIT – II:

Array and Modular Programming: Introduction to Function, Functions with Simple Output Parameters- Arrays: Declaring and Referencing Arrays, Array Subscripts, Using for Loops for Sequential Access

UNIT – III:

Structures, Unions, Strings, Pointers and files: Structures & Unions- definition- Pointers: Operations on Pointers –String handling - Text and data file processing.

UNIT – IV:

Evolution of OOP, OOP Paradigm, advantages of OOP, Comparison between functional programming and OOP Approach, characteristics of object oriented language. Introduction to C++, Identifier and keywords, constants, C++ operators, type conversion, Variable declaration, statements, expressions, input and output, Conditional expression, loop statements, breaking control statements.

UNIT – V:

Classes and objects, constructors and destructors, function and operator overloading, inheritance, manipulators, File streams, classes file modes.

REFERENCES:

1. E. Balagurusamy, “Programming in ANSI C”, Tata McGraw Hill, New Delhi, 8th Edition, 2019.
2. E. Balagurusamy, “Object-oriented Programming with C++”, Tata McGraw Hill, New Delhi, 7th Edition, 2017.

3. Herbert Schildt, “C++: The Complete Reference”, McGraw Hill Education India, 4th Edition, 2017.
4. Herbert Schildt, “C: The Complete Reference”, McGraw Hill Education India, 4th Edition, 2017.
5. https://www.w3schools.com/c/
6. https://www.khanacademy.org/computing/computer-programming

COURSE OUTCOMES:

The students would have gained knowledge of basics elements of C and C++ languages, their specifications, functions, passing of parameters, control constructs; Special features of C language such as Structures and Unions, Pointers and types of files and processing them; Classes and objects in C++ language and the features of C++ and to use them effectives to solve problems.

*****

First Year **BRIDGE COURSE II** **Semester I**

FUNDAMENTAL OF DATA STRUCTURES

Code: **(Theory)** **Credit: 4**

COURSE OBJECTIVES:

- To learn linear data structures – lists, stacks, and queues
- To learn different sorting and searching algorithms
- To understand Tree and Graph data structures

UNIT – I:

Abstract Data Types (ADTs): List ADT – array-based implementation – linked list implementation – singly linked lists- applications of lists: Polynomial Manipulation. Implementation of List ADT using an array and using a linked list in C.

UNIT – II:

STACKS and QUEUE: Stack ADT - Applications - Evaluating arithmetic expressions- Conversion of Infix to Postfix- Recursion. Queue ADT – Priority Queue - applications of queues. Implementation of Stack ADT and palindrome checking using C. Implementation of Queue operations using arrays in C.

UNIT – III:

SEARCHING AND SORTING ALGORITHMS :Divide and conquer methodology - Searching: Linear Search - Binary Search. Sorting: Insertion sort – Merge sort – Quick sort – Heap sort.

UNIT – IV:

Tree ADT – Tree traversals - Binary Tree ADT – expression trees – binary search tree ADT – applications of trees.

UNIT – V

GRAPHS: Definition – Representation of Graph – Breadth-first traversal - Depth-first traversal –

REFERENCES:

1. Mark Allen Weiss, “Data Structures and Algorithm Analysis in C”, 2nd Edition, Pearson Education,1997.
2. Brian W. Kernighan and Dennis M. Ritchie, “The C Programming Language”, 2nd Edition, Pearson Education, 1988.
3. Aho, Hopcroft and Ullman, “Data Structures and Algorithms”, Pearson Education, 1983.

4. S.Sridhar, “Design and Analysis of Algorithms”, First Edition, Oxford University Press. 2014
5. Byron Gottfried, Jitender Chhabra, “Programming with C” (Schaum’s Outlines Series), Mcgraw Hill Higher Ed., III Edition, 2010
6. YashvantKanetkar, “Data Structures Through C”, BPB publications, II edition, 2003.

COURSE OUTCOMES:

At the end of this course, the students will be able to:

- Implement lists and solve problems using them.
- Understand Stack and Queue operations
- Implement and apply trees and graphs to solve problems.
- Implement the various searching and sorting algorithms.
- Understand graphs

*****

First Year	**BRIDGE COURSE PRACTICAL I PROGRAMMING IN C AND C++**	**Semester I**
Code:	**(Practical)**	**Credit: 2**

Objectives:

To provide hands on training to handle various data types in C language; use control structures, strings and arrays; functions, pointers and their combinations. To define and create classes and objects, to understand the ideas of call by value, references; create and use all types of constructors and to use the ideas of inheritance, virtual functions and polymorphism. To learn to operate with files in C and C++ languages.

C-LANGUAGE:

1. Data types & Expressions, Constants & Variables, Operators, Operator Precedence and associatively, Storage Classes
2. Conditional statements, Looping Statements, Array and Modular Programming,
3. Basic Array programs using for loop, User defined functions, Recursion.
4. Programs on Two dimensional Arrays, Passing arrays as arguments, String handling based on String Functions and Character Operation

C++ - LANGUAGE:

1. Program using functions, functions with default arguments, implementation of call by value, address, reference.
2. Simple classes for understanding objects, member functions & constructors, classes with primitive data members, classes with arrays as data members, classes with pointers as data members, classes with constant data members, classes with static member functions.
3. Compile time polymorphism: operator overloading, function overloading,
4. File handling, sequential access, random access.

COURSE OUTCOMES:

The learners would have learnt to handle various data types in C language; use of control structures, strings and arrays; functions, pointers and their combinations; to define and create classes and objects, to understand the ideas of call by value, references; create and use all types of constructors and to use the ideas of inheritance, virtual functions and polymorphism; learnt to operate with files in C and C++ languages.

*****

First Year **CORE COURSE IV** **Semester II**

EMERGING TECHNOLOGIES IN DATA PROCESSING

Code: **(Theory)** **Credit: 5**

COURSE OBJECTIVES:

- Enhancing student's ability in dealing short-term dealing with day-to-day working capital decision
- Analyze and evaluate financial statements
- Develop knowledge on the allocation, management and funding of financial resources.

UNIT– I:

Database Systems Fundamentals: A Historical Perspective, Files System versus DBMS, Advantages of DBMS, Describing and storing data in a DBMS , Transaction management, Structure of a DBMS, People who work with Databases, Overview of Database Design. Entities, Attributes and Entity Sets, Relationships and Relationship sets, Additional Features of E-R Model: Key Constraints. Conceptual Design with the E-R Model. Data Storage & Indexing : File Organizations ,Organization of Records in Files, Indexing Structures, Primary & Secondary Indexes, Tree-structured Index, Hash-based Indexes, Multidimensional Indexes, Bitmap Indexes

UNIT– II:

Database System Architectures: Centralized and Client-Server Architectures, Server System Architectures, Parallel Systems, Distributed Systems, Parallel Databases, I/O Parallelism, Inter and Intra Query Parallelism, Intra and inter operation parallelism, Design of parallel systems, Distributed database concepts, Distributed Data storage, Distributed Transactions, Commit Protocols, Concurrency control, Distributed Query Processing.

UNIT– III:

RDBMS and SQL: Relational Query Languages, The SQL Query Language, Querying Multiple Relations, Creating Relations in SQL, Destroying and Altering Relations, Adding and Deleting Tuples, Integrity Constraints (ICs), Primary and Candidate Keys in SQL, Foreign Keys, Referential Integrity in SQL, Enforcing Referential Integrity, Categories of SQL Commands, Data Definition, Data Manipulation Statements: SELECT - The Basic Form Subqueries, Functions, GROUP BY Feature, Updating the Database, Data Definition Facilities, Views, Normalization: Functional Dependency, Anomalies in a Database, The normalization process: Conversion to first normal form, Conversion to second normal form, Conversion to third normal form, The boyce-code normal form(BCNF), Fourth Normal form and fifth normal form, normalization and database design, Denormalization

UNIT – IV:

Semi-Structured Data: XML database management system.XML databases, XML schema, Storing XML in Databases, XML and SQL. XML Query processing: XML query languages, XQuery, XPath. Approaches for XML query processing, Query processing on relational structure and storage schema.

UNIT – V:

No SQL Databases: Column-oriented Databases, Graph Databases, Key-value pair Databases, Document Databases. CAP Theorem, Sharding. **Big Data Management**: Hadoop: HDFS, Dealing with Massive Datasets-Map Reduce and Hadoop.

UNIT – VI CURRENT CONTOURS (For continuous internal assessment only):

Contemporary Developments Related to the Course during the Semester Concerned

REFERENCES:

1. A Silberschatz, H Korth, S Sudarshan, "Database System and Concepts", fifth Edition McGraw-Hill
2. Rob, Coronel, "Database Systems", Seventh Edition, Cengage Learning.
3. Guy Harrison, "Next Generation Data Bases – NoSQL, NewSQL and Big Data", 1stEd ,Apress, 2015.
4. Authored by DT Editorial Services , "Big Data, Black Book: Covers Hadoop 2, MapReduce, Hive, YARN, Pig, R and Data Visualization WileyIndia, 2016
5. Ramakrishna R. &Gehrke J, Database Management Systems, 3e, Mc-Graw Hill, 2003.
6. Elmarsi R, &Navathe S B, Fundamental of Database System, 5e, Pearson Education, 2008.
7. Robinson, I, Webber, J, &Eifrem E, Graph Databases, 2e, O'Reilly, 2015.
8. https://docs.snowflake.com/en/user-guide/semistructured-concepts.html
9. https://www.w3resource.com/mongodb/nosql.php
10. https://www.tutorialspoint.com/sql/sql-rdbms-concepts.htm
11. https://www.javatpoint.com/dbms-normalization

COURSE OUTCOMES:

After completion of this course, students will be able to

- Employ ER diagram as a data modeling technique to represent entity framework.
- Compare the architectures of distributed and parallel systems.
- Experiment with SQL queries and construct normalized databases
- Demonstrate the semi-structured data handling using XML and JSON
- Explain the types of NoSQL databases and Map reduce framework.

First Year **CORE COURSE V** **Semester II**

ADVANCED OPERATING SYSTEMS

Code: **(Theory)** **Credit: 5**

COURSE OBJECTIVES:

- To study the characteristics of Multiprocessor and Multi computer
- To understand the advance concepts of distributed operating systems
- To get an insight into the various issues and solutions in distributed operating systems

UNIT – I:

Multiprocessor Operating Systems: System Architectures- Structures of OS – OS design issues –Process synchronization – Process Scheduling and Allocation- memory management.

UNIT – II:

Distributed Operating Systems: System Architectures- Design issues – Communication models –clock synchronization – mutual exclusion – election algorithms- Distributed Deadlock detection

UNIT – III:

Distributed scheduling - Distributed shared memory - Distributed File system – Multimedia file systems - File placement - Caching

UNIT – IV:

Database Operating Systems: Requirements of Database OS – Transaction process model – Synchronization primitives - Concurrency control algorithms

UNIT – V:

Mobile Operating Systems: ARM and Intel architectures - Power Management - Mobile OS Architectures - Underlying OS - Kernel structure and native level programming – Runtime issues- Approaches to power management

UNIT – VI CURRENT CONTOURS (For continuous internal assessment only):

Contemporary Developments Related to the Course during the Semester Concerned

REFERENCES:

1. MukeshSinghal and Niranjan G. Shivaratri, “Advanced Concepts in Operating Systems Distributed, Database, and Multiprocessor Operating Systems”, Tata McGraw-Hill, 2001

2. A S Tanenbaum, Distributed Operating Systems, Pearson Education Asia, 2001
3. Source Wikipedia, Mobile Operating Systems, General Books LLC, 2010
4. Abraham Silberschatz, Peter B. Galvin, Greg Gagne, "Operating System Concepts", Wiley, Eighth Edition, 2008.
5. https://www.javatpoint.com/os-tutorial
6. https://www.khanacademy.org/computing/computer-science
7. http://www.uobabylon.edu.iq/download/M.S%202013-2014/Operating_System_Concepts,_8th_Edition%5BA4%5D.pdf
8. http://index-of.es/Varios-2/Modern%20Operating%20Systems%204th%20Edition.pdf

COURSE OUTCOMES:

At the end of the course, the students will be able to

- Knowledge about advance concepts in OS
- Demonstrate the various issues in distributed operating systems
- Identify the different features of data base operating systems
- Understand back end operating system
- Understand Mobile operating system

*****

First Year	CORE COURSE VI COMPUTER GRAPHICS AND ANIMATION	Semester II
Code:	(Theory)	Credit: 5

COURSE OBJECTIVES:

To impart knowledge to make the students

- To learn basic understanding of Computer Graphics
- To get clear idea about various graphic algorithms.
- To understand the 2D and 3D transformations, models and generation techniques

UNIT - I:

Output Primitives: Points and Lines, Line-Drawing Algorithms: DDA Algorithm, Bresenham's Line Algorithm, Line Function, Circle Generation Algorithms, Ellipse Generation AlgorithmsAttributes of output Primitives: Line Attributes, Color and Gray Scale levels, Area Fill Attributes, Character Attributes, Bundled Attributes, Antialiasing.

UNIT - II:

Two Dimensional Geometric Transformations: Basic Transformations, Matrix Representation and Homogenous Coordinates, Composite Transformations, Other Transformations.

Two Dimensional Viewing: The Viewing pipeline, Viewing Coordinates Reference Frame, Window to Viewport Coordinate Transformations, Two Dimensional Viewing Functions, Clipping Operations, Point Clipping,

Line Clipping: Cohen-Sutherland Line Clipping, Polygon Clipping: Sutherland-Hodgeman Polygon Clipping.

UNIT - III:

Three Dimensional Concepts: Three Dimensional Display Methods. Three Dimensional Object Representations: Polygon Surfaces, Quadric Surfaces, Superquadrics.Three Dimensional Geometric and Modeling Transformations: Translation, Rotation, Scaling, Other Transformations, Composite Transformations, Three Dimensional Transformation Functions.

UNIT - IV:

Animation: Introduction to Animation – Principles of Animation - Pipeline – Moving Camera Character - Designing and Framework for View Dependent Animation – The View Space – Distance of Viewpoint

UNIT - V:

Dependent Animation: View Dependent Animation from Sketches – Overview of pipeline – Inputs – Recovering the Camera – Posing the Character – Animating the Character- View Dependent Animation from Multimodal Inputs – Challenges in Multimodal Authoring of Animation – Creating a View Space from Video

UNIT VI Current Contours (For continuous internal assessment only)
Contemporary Developments Related to the Course during the Semester Concerned

REFERENCES:

1. Donald Hearn and Pauline Baker M, "Computer Graphics", Prentice Hall, New Delhi, 2007
2. ParagChaudhuri, PremKalra and Subhashis Banerjee, "View Dependent Character Animation", Springer-Verlag London Limited, 2007
3. Foley, Vandam, Feiner and Hughes, ―Computer Graphics: Principles and Practice‖, 2nd Edition, Pearson Education, 2003.
4. Jeffrey McConnell, ―Computer Graphics: Theory into Practice‖, Jones and Bartlett Publishers, 2006.
5. Hill F S Jr., "Computer Graphics", Maxwell Macmillan , 1990.
6. Anatomy of the Artist – Thompson & Thompson (Recent Edition)
7. https://www.blender.org/support/tutorials/
8. https://www.docme.su/doc/1765678/parag-chaudhuri--prem-kalra--subhashis-banerjee---view-de...
9. https://www.javatpoint.com/computer-graphics-tutorial
10. https://www.geeksforgeeks.org/computer-graphics-2/

COURSE OUTCOMES:

At the end of the course, students will be able to:

- Develop software tools such as games and animation
- Create interactive computer graphics using OpenGL
- Understand a typical graphics pipeline and made pictures with their computer.
- Understand the Multimedia animation and Desktop Computing
- Develop skills on animation drawing tools

*****

First Year	CORE PRACTICAL III OPERATING SYSTEMS	Semester II
Code:	(Practical)	Credit: 2

LIST OF EXPERIMENTS (Using C/C++/Java):

1. Simulate the following CPU Scheduling algorithms

   a) FCFS b) SJF c) Round Robin d) priority

2. Write programs using the I/O system calls of UNIX/LINUX operating system (open, read, write, close, fcntl, seek, stat, opendir, readdir).

3. Simulate Bankers Algorithm for Deadlock Avoidance and Prevention.

5. Implement the Producer – Consumer problem using semaphores using UNIX/LINUX system calls.

6. Illustrate the following IPC mechanisms

   a) Pipes b) FIFOs c) Message Queues d) Shared Memory

7. Simulate the following memory management techniques

   a) Paging b) Segmentation

*****

First Year	CORE CHOICE PRACTICAL IV COMPUTER GRAPHICS AND ANIMATION	Semester II
Code:	(Practical)	Credit: 2

LIST OF EXPERINETNS(Lab can be conducted in "C" language / Virtual Labs /Open GL)

1. Digital differential Analyzer
2. Line Drawing Algorithms
3. Mid-point Circle Generation Algorithm
4. Creating two-Dimensional Objects
5. Two-dimensional Transformation
6. Picture Coloring
7. Three-Dimensional transformation
8. Simple Animation using Transformation
9. Key-Frame Animation
10. Design Animation using FLASH

*****

First Year **ELECTIVE COURSE II** **Semester II**

INTERNET OF THINGS (IoT)

Code: **(Theory)** **Credit: 4**

COURSE OBJECTIVES:

- To gain knowledge on bases of Internet of Things (IoT)
- To understand IoT Architecture and the Protocols related to IoT
- To acquire knowledge about WoT

UNIT – I:

INTRODUCTION To IoT: Internet of Things - Physical Design- Logical Design- IoT Enabling Technologies - IoT Levels and Deployment Templates - Domain Specific IoTs - IoT and M2M - IoT System Management with NETCONF-YANG- IoT Platforms Design Methodology.

UNIT – II:

IoT ARCHITECTURE: M2M high-level ETSI architecture - IETF architecture for IoT - OGC architecture - IoT reference model - Domain model - information model - functional model - communication model - IoT reference architecture

UNIT – III:

IoT PROTOCOLS: Protocol Standardization for IoT – Efforts – M2M and WSN Protocols – SCADA and RFID Protocols – Unified Data Standards – Protocols – IEEE 802.15.4 – BACNet Protocol – Modbus– Zigbee Architecture – Network layer – 6LowPAN - CoAP - Security

UNIT – IV:

WEB OF THINGS: Web of Things versus Internet of Things – Two Pillars of the Web – Architecture Standardization for WoT– Platform Middleware for WoT – Unified Multitier WoT Architecture – WoT Portals and Business Intelligence. Cloud of Things: Grid/SOA and Cloud Computing – Cloud Middleware – Cloud Standards – Cloud Providers and Systems – Mobile Cloud Computing – The Cloud of Things Architecture.

UNIT – V:

APPLICATIONS: The Role of the Internet of Things for Increased Autonomy and Agility in Collaborative Production Environments - Resource Management in the Internet of Things: Clustering, Synchronisation and Software Agents. Applications - Smart Grid – Electrical Vehicle Charging.

UNIT – VI CURRENT CONTOURS (For continuous internal assessment only):

Contemporary Developments Related to the Course during the Semester Concerned

RESOURCES:

1. ArshdeepBahga, Vijay Madisetti, “Internet of Things – A hands-on approach”, Universities Press, 2015.
2. Dieter Uckelmann, Mark Harrison, Michahelles, Florian (Eds), “Architecting the Internet of Things”, Springer, 2011.
3. Jan Ho¨ ller, VlasiosTsiatsis, Catherine Mulligan, Stamatis ,Karnouskos, Stefan Avesand. David Boyle, "From Machine-to-Machine to the Internet of Things - Introduction to a New Age of Intelligence", Elsevier, 2014.
4. Networks, Crowds, and Markets: Reasoning About a Highly Connected World - David Easley and Jon Kleinberg, Cambridge University Press - 2010.
5. Olivier Hersent, David Boswarthick, Omar Elloumi , “The Internet of Things – Key applications and Protocols”, Wiley, 2012.
6. https://www.javatpoint.com/iot-internet-of-things
7. https://www.geeksforgeeks.org/introduction-to-internet-of-things-iot-set-1/
8. https://www.edureka.co/blog/iot-tutorial/

COURSE OUTCOMES:

At the end of this course, students should be able to

- Gain the basic knowledge about IoT and they will be able to use IoT related products in real life.
- Acquire knowledge about IoT architecture
- Understand IoT protocols.
- Helps to understand the concept of the Web of Thing.
- Understand the application areas of the IoT.

*****

First Year	BRIDGE COURSE III WEB DESIGN	Semester II
Code:	(Theory)	Credit: 4

COURSE OBJECTIVES:

To understand the basic concepts of Internet; identify the features of HTML tags; to design the HTML tables, frames and forms; to design applications with JavaScript Programming; to comprehend the objects in HTML and Java Script.

UNIT – I:

Networking Concepts: INTERNET - History - Applications-Users – Protocols - Host Machines and Host Names - Internet Architecture and Packet Switching-Client Server Model - Band width and Asynchronous Communication. Connection: Dial-up Access-Direct and Dedicated Connections - shell or TCP/ IP accounts - Domains and Addresses – IP addresses.

UNIT – II:

HTML: Introduction to HTML Tags - Document Layout - Comments - Headings-Paragraphs -Breaks - Texts - Lists - Special Characters.

UNIT – III:

HTML: Tables - Linking documents - Frames - Form and its elements.

UNIT – IV:

JavaScript: Introduction to JavaScript - JavaScript in web pages-writing JavaScript with HTML - Basic programming techniques - operators and expressions - conditional checking - loops - functions - user defined functions - dialog boxes.

UNIT – V:

JavaScript: JavaScript DOM: JSS DOM - understanding objects in HTML - browser objects - web page object hierarchy - Handling events - The form object - built-in objects-user defined objects - cookies - setting a cookie.

REFERENCES:

1. Wendy G. Lehnert, “Internet 101 - A Beginners Guide To The Internet And The World Wide Web”, Addison-Wesley, 1999 (Unit-I).
2. Ivan N. Bayross, “Web enabled Commercial Application Development using HTML, JavaScript, DHTML and PHP”, 4th Revised Edition, BPB Publications, New Delhi, 2010.(Units-II, III, IV, V)
3. Chuck Musciano& Bill Kennedy, “HTML - The Definitive Guide”, Shroff Publishers & Distributors Pvt. Ltd., Calcutta - 1999.

4. Raj Kamal, “Internet And Web Technologies”, TMH, New Delhi, SBN: 9780070472969
5. John Pollock, “JavaScript A Beginner’s Guide”, The McGraw-Hill, 2010.
6. https://www.w3schools.com/
7. https://javascript.info/

COURSE OUTCOMES:

After completing this course the students will be able to understand the basic concepts of Internet; identify the features of HTML tags; design the HTML tables, frames and forms; design applications with JavaScript Programming; work with the objects in HTML and Java Script.

*****

First Year	BRIDGE COURSE IV DIGITAL ELECTRONICS	Semester II
Code:	(Theory)	Credit: 4

COURSE OBJECTIVES:

- To introduce the concept of IC logic families
- To Understand digital principles,
- To gain knowledge about Boolean Algebra, logic gates, combinational circuits and sequential circuits.

Unit –I

IC Logic families: IC logic families- definition. General characteristics- TTL, ECL and CMOS, advantages and disadvantages, Definition- Tri-state logic. IC- definition, advantages of IC over discrete components.

Unit –II

Digital Principles: Definitions- bit, nibble, byte, word, and parity bit. Number system definition, types, radix, decimal, BCD, binary and hexadecimal. BCD addition. Binary addition, subtraction, Multiplication, Division, 1's and 2's complement. Hexadecimal addition, subtraction, advantages. Conversion- decimal to binary and hexadecimal and vice versa.ASCII, Gray codes, and list applications.

Unit –III

Boolean Algebra & Logic Gates: Definition- Boolean variable, complement, Boolean function, expression, truth table and Buffer.Boolean Algebra- rules and laws. Logic gates NOT, AND, OR, NAND, NOR, EX-OR- definition, symbol, Boolean equation, truth table and working. De Morgan's theorems- statement and equations. Universal gates- definition,
realisation of NOT, OR, AND and EXOR gates.

Unit –IV

Combinational Logic Circuits: Definition. Adders- definition, types. Half adder- block diagram, logic diagram using AND and XOR, truth table and working. Full adder- block diagram, logic diagram using AND, OR and XOR, truth table and working.

Unit –V

Sequential Logic Circuits: Definitions- level and edge triggering. Flip flops definition, types and applications. RS flip flop and clocked RS flip flop- block diagram, truth table, logic diagram using NAND gates and working. JK flip flop- block diagram, truth table, logic diagram using NAND gates and working.

REFERENCES:

1. Digital Fundamentals by T. L. Floyd, Pearson International Publications, Ninth Edition, 2000.
2. Principles of Digital Electronics By K. Meena, PHI Learning Pvt. Ltd,New Delhi.2009
3. Electronics Principles by Malvino and Leach, Mc. Graw Hill, Third edition. 2000.
4. Modern Digital Electronics by R P Jain, Tata McGraw-Hill Education, 2003.
5. Digital Electronics: Principles and Applications by R. L. Tokheim, Tata McGraw-Hill Education, 2013.
6. Electronics Analog and Digital by I. J. Nagrath, PHI Learning Pvt. Ltd., 2013 Edition.
7. Principles of Digital Electronics by K. Meena, PHI Learning Pvt. Ltd., FourthPrinting, 2013.
8. https://en.wikipedia.org/wiki/
9. 2.https://www.google.co.in/search?sclient=psyab&site=&source=hp&btnG=Search&q=JK+flip+flop+using+NAND+gates
10. www.electronics-tutorials.ws › Sequential Logic
11. www.circuitstoday.com/flip-flops

COURSE OUTCOMES:

- 1.Understand the basics of IC logic families.
- Appraise digital principles and number system conversion.
- Reveal the Logic gates
- Gain knowledge about different Combinational logic circuits.
- Understand various Sequential logic circuits

*****

First Year	**BRIDGE COURSE PRACTICAL II WEB DESIGN**	**Semester II**
Code:	**(Practical)**	**Credit: 2**

COURSE OBJECTIVES:

To have hands-on experience of displaying formatted text and pages, learn to accept input from user and display using tables and frames; to know how to create forms in web pages to collect data.

1. Text formatting
2. Getting input and performing string manipulation operations
3. Using tables for neatly displaying information about an organization
4. Using frames to categories and display information in a easy-to-understand format.
5. Using forms to create web pages for applying for a position in an organization
6. Event handling
7. Creating and managing cookies

*****

Second Year	CORE COURSE VII COMPILER DESIGN	Semester III
Code:	(Theory)	Credit: 5

COURSE OBJECTIVES:

- Discover principles, algorithms and techniques that can be used to construct various phases of compiler.
- Acquire knowledge about finite automata and regular expressions
- Explore knowledge about Syntax Directed definitions and translation scheme

UNIT - I LEXICAL ANALYSIS:

Language Processors, The Structure of a Compiler, Parameter passing mechanism – Symbol table - The role of the lexical analyzer - Input buffering - Specification of tokens - Recognition of tokens – Finite automata - Regular expression to automata.

UNIT – II SYNTAX ANALYSIS:

The role of the parser - Context-free grammars - Writing a grammar - Top down Parsing - Bottom-up Parsing - LR parsers- LALR parsers.

UNIT – III SEMANTIC ANALYSIS:

Inherited and Synthesized attributes – Dependency graphs – Ordering the evaluation of attributes – S-attributed definitions – L-attributed definitions – Applications of Syntax Directed translation – Syntax Directed translations schemes - Storage organization – Stack allocation of space.

UNIT – IV INTERMEDIATE CODE GENERATION:

Variants of Syntax trees – Three Address code – Types and Declarations - Translation of Expressions – Type checking - Control flow - Back patching - Switch Statements - Procedure calls.

UNIT – V CODE GENERATION AND CODE OPTIMIZATION:

Issues in the design of a code generator - The target language – Address in the Target Code – Basic Block and Flow graphs – Optimization of Basic Blocks - A simple code generator – Peephole Optimization.

UNIT – VI CURRENT CONTOURS (For continuous internal assessment only):

Contemporary Developments Related to the Course during the Semester Concerned

REFERENCES:

1. Alfred V. Aho, Monica S.Lam, Ravi Sethi and Jeffrey D. Ullman, "Compilers-Principles, Techniques and Tools", Second Edition, Pearson Education Asia, 2009.
2. A.V. Aho, Ravi Sethi, J.D. Ullman, Compilers - Principles, Techniques and Tools, Addison- Wesley, 2003.
3. Fischer Leblanc, Crafting Compiler, Benjamin Cummings, Menlo Park, 1988.
4. KennathC.Louden, Compiler Construction Principles and Practice, Vikas publishing House, 2004.
5. Allen I. Holub, Compiler Design in C, Prentice Hall of India, 2001.
6. S.GodfreyWinster, S.Aruna Devi, R.Sujatha, "Compiler Design", yesdee Publishers, Third Reprint 2019.
7. https://www.javatpoint.com/compiler-tutorial
8. https://www.geeksforgeeks.org/compiler-design-tutorials/

COURSE OUTCOMES:

On the successful completion of this course, Students will be able to:

- Understand the basic principles of compiler design.
- Learn context free grammars, compiler parsing techniques.
- Use the knowledge of patterns, tokens & regular expressions for solving a problem in the field of data mining.
- Specify and analyse the lexical, syntactic and semantic structures of advanced language features.
- Separate the lexical, syntactic and semantic analysis into meaningful phases for a compiler to undertake language translation.

*****

Second Year **CORE COURSE VIII** **Semester III**

MACHINE LEARNING TECHNIQUES

Code: **(Theory)** **Credit: 5**

COURSE OBJECTIVES:

- To Learn about Machine Intelligence and Machine Learning applications
- To understand the theoretical and practical aspects of Probabilistic Graphical Models
- To understand how to perform evaluation of learning algorithms and model selection

UNIT – I INTRODUCTION:

Machine Learning - Machine Learning Foundations –Overview – Design of a Learning system - Types of machine learning –Applications Mathematical foundations of machine learning - random variables and probabilities - Probability Theory – Probability distributions -Decision Theory- Bayes Decision Theory - Information Theory

UNIT – II SUPERVISED LEARNING:

Linear Models for Regression - Linear Models for Classification – Naïve Bayes - Discriminant Functions -Probabilistic Generative Models -Probabilistic Discriminative Models - Bayesian Logistic Regression. Decision Trees - Classification Trees- egression Trees - Pruning. Neural Networks -Feed-forward Network Functions - Back- propagation. Support vector machines - Ensemble methods- Bagging- Boosting

UNIT – III UNSUPERVISED LEARNING:

Clustering- K-means - EM Algorithm- Mixtures of Gaussians. The Curse of Dimensionality Reduction - Factor analysis - Principal Component Analysis - Probabilistic PCA- Independent components analysis

UNIT – IV PROBABILISTIC GRAPHICAL MODELS:

Graphical Models - Undirected graphical models - Markov Random Fields - Directed Graphical Models -Bayesian Networks - Conditional independence properties - Inference – Learning- Generalization - Hidden Markov Models - Conditional random fields(CRFs)

U**NIT – V ADVANCED LEARNING:**

Sampling –Basic sampling methods – Monte Carlo.Reinforcement Learning- K-Armed Bandit-Elements - Model-Based Learning- Value Iteration- Policy Iteration. Temporal Difference Learning- Exploration Strategies- Deterministic and Non-deterministic Rewards and Actions Computational Learning Theory - Mistake bound analysis, sample complexity analysis, VC dimension. Occam learning, accuracy and confidence boosting

UNIT – VI CURRENT CONTOURS (For continuous internal assessment only):

Contemporary Developments Related to the Course during the Semester Concerned

REFERENCES:

1. Christopher Bishop, "Pattern Recognition and Machine Learning" Springer, 2007.
2. Kevin P. Murphy, "Machine Learning: A Probabilistic Perspective", MIT Press, 2012.
3. EthemAlpaydin, "Introduction to Machine Learning", MIT Press, Third Edition, 2014.
4. Tom Mitchell, "MachineLearning", McGraw-Hill, 1997.
5. Trevor Hastie, Robert Tibshirani, Jerome Friedman, "The Elements of Statistical Learning", Springer, Second Edition, 2011.
6. Stephen Marsland, "Machine Learning - An Algorithmic Perspective", Chapman and Hall/CRC Press, Second Edition, 2014.
7. Demystifying Machine Learning, Neural Networks and Deep Learning By Suresh Samudrala · 2019, Notion Press
8. Machine Learning, By Rajiv Chopra KHANNA PUBLISHING HOUSE,2020
9. https://data-flair.training/blogs/machine-learning-tutorial/
10. https://www.cs.ubc.ca/~murphyk/Bayes/bnintro.html
11. https://www.geeksforgeeks.org/machine-learning/

COURSE OUTCOMES:

At the end of the course, the students will be able to:

- Have a good understanding of the fundamental issues and challenges of machine learning:
- Have an understanding of the strengths and weaknesses of many popular machine learning approaches.
- Be able to design and implement various machine learning algorithms in a range of real-world applications.
- Use a tool to implement typical clustering algorithms for different types of applications
- Design and implement an HMM for a sequence model type of application

*****

Second Year **CORE COURSE IX** **Semester III**

BIG DATA ANALYTICS

Code: **(Theory)** **Credit: 5**

COURSE OBJECTIVES:

- To provide grounding in basic and advanced methods to big data technology and tools.
- To gain knowledge about MapReduce and Hadoop and its ecosystem.
- To understand Advanced analytical theory and methods.

UNIT – I INTRODUCTION TO BIG DATA ANALYTICS:

Big Data Overview – Data Structures – Analyst Perspective on Data Repositories - State of the Practice in Analytics – BI Versus Data Science - Current Analytical Architecture – Drivers of Big Data – Big Data Ecosystem - Data Ana lytics Lifecycle – Data Discovery – Data Preparation – Model Planning – Model Building – Communicate Results – Operationalize.

UNIT – II BASIC DATA ANALYTIC METHODS USING R:

Introduction to R programming – R Graphical User Interfaces – Data Import and Export – Attribute and Data Types – Descriptive Statistics Exploratory Data Analysis : Visualization Before Analysis – Dirty Data – Visualizing a Single Variable – Examining Multiple Variables Data Exploration Versus Presentation -– Statistical Methods of Evaluation : Hypothesis Testing – Difference of Means – Wilcoxon Rank-Sum Test – Type I and Type II Errors – Power and Sample Size – ANOVA..

UNIT – III ADVANCED ANALYTICAL THEORY AND METHODS:

Clustering – K Means – Use Cases – Overview – Determining number of clusters – Diagnostics – Reasons to choose and cautions – Additional Algorithms - Association Rules: A Priori Algorithm – Evaluation of Candidate Rules – Applications of Association Rules – Validation and Testing – Diagnostics. Regression: Linear Regrression and Logistic Regression :– Use cases – Model Description – Diagnostics - Additional Regression Models.

UNIT – IV CLASSIFICATION:

Decision Trees – Overview – Genetic Algorithm – Decision Tree Algorithms – Evaluating Decision Tree – Decision Trees in R - Na'ive Bayes – Bayes Theorem – Naïve Bayes Clasifier – Smoothing – Diagnostics – Naïve Bayes in R – Diagnostics of Classifiers – Additional Classification Methods - Time Series Analysis : Overview – Box – Jenkins Methodology – ARIMA Model – Autocorrelation Function – Autoregressive Models – Moving Average Models – ARMA and ARIMA Models – Building and Evaluating and ARIMA Model - Text Analysis : Text Analysis Steps – Example – Collecting – Representing Term Frequency – Categorizing – Determining Sentiments – Gaining Insights.

UNIT – V ADVANCED ANALYTICS-TECHNOLOGY AND TOOLS:

MapReduce and Hadoop : Analytics for Unstructured Data .- *UseCases* - *MapReduce*- Apache Hadoop – The Hadoop Ecosystem – pig – Hive – Hbase – Manout – NoSQL - Tools in Database Analytics : SQL Essentials – Joins – Set operations – Grouping Extensions.

UNIT – VI CURRENT CONTOURS (For continuous internal assessment only):

Contemporary Developments Related to the Course during the Semester Concerned.

REFERENCES:

1. Data Science & Big Data Analytics: Discovering, Analyzing, Visualizing and Presenting Data", EMC Education Services, John Wiley & Sons, Inc. 2015.
2. Noreen Burlingame, "The little book on Big Data", New Street publishers, 2012.
3. Anil Maheshwari, "Data Analytics", McGraw Hill Education, 2017.
4. Norman Matloff, "The Art of R Programming: A Tour of Statistical Software Design", Starch Press; 1 edition, 2011.
5. SandipRakshit, "R for Beginners", McGraw Hill Education, 2017.
6. http://www.johndcook.com/R_language_for_programmers.html.
7. http://bigdatauniversity.com/.
8. http://home.ubalt.edu/ntsbarsh/stat-data/topics.htm#rintroduction.
9. https://www.guru99.com/bigdata-tutorials.html
10. https://www.javatpoint.com/hadoop-tutorial
11. https://www.udemy.com/course/big-data-and-hadoop-framework/

COURSE OUTCOMES:

At the end of the course, the students will be able to:

- Apply Hadoop ecosystem components.
- Participate data science and big data analytics projects
- Understand the core objective of the Big Data Framework is to provide a structure for enterprise organizations that aim to benefit from the potential of Big Data.
- Understand the Big Data that is more than just the combination of skilled people and technology – it requires structure and capabilities.
- Gain the knowledge about Technology and Tools for Advanced Analytics.

*****

Second Year **CORE PRACTICAL V** **Semester III**

MACHINE LEARNING TECHNIQUES

Code: **(Practical)** **Credit: 2**

1. Given a dataset. Write a program to compute the Covariance, Correlation between a pair of attributes. Extend the program to compute the Covariance Matrix and Correlation Matrix.
2. Write a program to demonstrate the working of the decision tree based ID3 algorithm. Use an appropriate data set for building the decision tree and apply this knowledge to classify a new sample.
3. Write a program to implement k-Nearest Neighbour algorithm to classify the iris data set. Print both correct and wrong predictions. Python ML library classes can be used for this problem.
4. Write a program to implement feature reduction using Principle Component Analysis
5. Write a program to implement the naïve Bayesian classifier for a sample training data set stored as a .CSV file. Compute the accuracy of the classifier, considering a few test data sets.
6. Given a dataset for classification task. Write a program to implement Support Vector Machine and estimate its test performance.
7. Build an Artificial Neural Network by implementing the Back propagation algorithm and test the same using appropriate data sets.
8. Write a program to implement K means clustering algorithm. Select your own dataset to test the program. Demonstrate the nature of output with varying value of K.

*****

Second Year	**CORE PRACTICAL VI** **BIG DATA ANALYTICS**	**Semester III**
Code:	**(Practical)**	**Credit: 2**

List of Experiments

1. To get the input from user and perform numerical operations (MAX, MIN, AVG, SUM, SQRT, ROUND) using R.
2. To perform data import/export (.CSV, .XLS, .TXT) operations using data frames in R.
3. To get the input matrix from user and perform Matrix addition, subtraction, multiplication, inverse transpose and division operations using vector concept in R.
4. To perform Association Rule Mining and Clustering using R.
5. To perform data pre-processing operations i) Handling Missing data ii) Min-Max normalization
6. To perform Simple Linear Regression with R.
7. To perform market basket analysis using Association Rules (Apriori).
8. Using R perform the Time-series analysis with respect to stock market data

*****

Second Year **ELECTIVE COURSE III** **Semester III**

NATURAL LANGUAGE PROCESSING

Code: **(Theory)** **Credit: 4**

COURSE OBJECTIVES:

- To familiarize the concepts and techniques of Natural language Processing for analyzing words based on Morphology and CORPUS.
- Torelate mathematical foundations, Probability theory with Linguistic essentials such as syntactic and semantic analysis of text.
- To apply the Statistical learning methods and cutting-edge research models from deep learning.

UNIT – I INTRODUCTION TO NLP:

Various stages of NLP –The Ambiguity of Language: Why NLP Is Difficult- Parts of Speech: Nouns and Pronouns, Words: Determiners and adjectives, verbs, Phrase Structure. Statistics Essential Information Theory: Entropy, perplexity, The relation to language, Cross entropy

UNIT – II TEXT PREPROCESSING AND MORPHOLOGY:

Character Encoding, Word Segmentation, Sentence Segmentation, Introduction to Corpora, Corpora Analysis. Inflectional and Derivation Morphology, Morphological analysis and generation using Finite State Automata and Finite State transducer.

UNIT – III LANGUAGE MODELLING:

Words - Collocations- Frequency-Mean and Variance –Hypothesis testing: The t test, Hypothesis testing of differences, Pearson's chi-square test, Likelihood ratios. Statistical Inference: n -gram Models over Sparse Data: Bins: Forming Equivalence Classes- N gram model - Statistical Estimators- Combining Estimators

UNIT – IV WORD SENSE DISAMBIGUATION:

Methodological Preliminaries, Supervised Disambiguation: Bayesian classification, An information- theoretic approach, Dictionary-Based Disambiguation: Disambiguation based on sense, Thesaurus- based disambiguation, Disambiguation based on translations in a second-language corpus.

UNIT – V SYNTAX AND SEMANTICS:

Shallow Parsing and Chunking, Shallow Parsing with Conditional Random Fields (CRF), Lexical Semantics, WordNet, Thematic Roles, Semantic Role Labelling with CRFs. Statistical Alignment and Machine Translation, Text alignment, Word alignment, Information extraction, Text mining, Information Retrieval, NL interfaces, Sentimental Analysis, Question Answering Systems, Social network analysis.

UNIT VI CURRENT CONTOURS (For continuous internal assessment only):

Contemporary Developments Related to the Course during the Semester Concerned

REFERENCES:

1. Christopher D. Manning and HinrichSchutze, " Foundations of Natural Language Processing" , 6th Edition, The MIT Press Cambridge, Massachusetts London, England, 2003
2. Daniel Jurafsky and James H. Martin "Speech and Language Processing", 3rd edition, Prentice Hall, 2009.
3. NitinIndurkhya, Fred J. Damerau "Handbook of Natural Language Processing", Second Edition, CRC Press, 2010.
4. James Allen "Natural Language Understanding", Pearson Publication 8th Edition. 2012.
5. Chris Manning and HinrichSchütze, "Foundations of Statistical Natural Language Processing", 2nd edition, MITPress Cambridge, MA, 2003.
6. Hobson lane, Cole Howard, HannesHapke, "Natural language processing in action" MANNING Publications, 2019.
7. Alexander Clark, Chris Fox, Shalom Lappin, "The Handbook of Computational Linguisticsand Natural Language Processing", Wiley-Blackwell, 2012
8. Rajesh Arumugam, RajalingappaShanmugamani "Hands-on natural language processing with python: A practical guide to applying deep learning architectures to your NLP application". PACKT publisher, 2018.
9. https://www.tutorialspoint.com/natural_language_processing/index.htm
10. https://www.javatpoint.com/nlp
11. https://developer.ibm.com/technologies/natural-language-processing/tutorials

COURSE OUTCOMES:

- Apply the principles and Process of Human Languages such as English and other Indian Languages using computers.
- Realize semantics and pragmatics of English language for text processing
- Understand the Language Modelling
- Demonstrate the state-of-the-art algorithms and techniques for text-based processing of natural language with respect to morphology.
- Understand text preprocessing techniques.

*****

Second Year	BRIDGE COURSE V PYTHON PROGRAMMING	Semester III
Code:	(Theory)	Credit: 4

COURSE OBJECTIVES:

- To enable the students to understand the concepts of programming in Python. To provide knowledge in core python, concepts like modules and packages, file handing, regular expressions, exception handling, to be able to work with numbers and data and to use them with visualisation tools.

UNIT – I INTRODUCTION:

Welcome to Python - Origins – Features of Python –Downloading and Installing Python –Running Python - Comments - Operators – Variables and Assignment - Numbers –Strings – Lists and Tuples –Dictionaries - Python Objects: Standard Types - Other Built-in Types – Internal Types.

UNIT – II NUMBERS:

Introduction to Numbers - Integers - Floating Point Numbers – Complex Numbers – Operators -Built-in and Factory Functions. Conditionals and Loops: if statement - else statement - else if statement - while statement - for statement – break statement - continue statement - pass statement.

UNIT – III SEQUENCES (STRINGS, LISTS AND TUPLES):

Strings: String and Operators- String Only Operators – String Built-in Methods – Special features of Strings - Lists: Operators –Built in functions- List type Built-in functions-Special features of Lists - Tuples: Tuple operators and Built-in function-Special features of Tuples.

UNIT – IV FILE I/O AND FUNCTIONS:

File Objects –File Built-in Functions-File Built-in Methods-File Built-in Attributes-Standard Files-Command-Line Arguments-

UNIT – V FUNCTIONS:

Calling, Creating and Passing functions-Formal and variable length arguments-recursion

UNIT – VI CURRENT CONTOURS (For continuous internal assessment only):

Contemporary DevelopmentsRelatedtotheCourseduringtheSemesterConcerned.

REFERENCES:

1. Wesley J Chun, Core Python Programming, 2nd Edition, Prentice Hall Publisher, 2006

   Unit I: Chapter 1,2,3,4 Unit II: Chapter 5, 8
   Unit III: Chapter 6 Unit IV: Chapter 9,11
   Unit V: Chapter 13
2. https://docs.python.org/3/tutorial/

COURSE OUTCOMES:

At the end of the course, students will be able to:

- Understand the building blocks of python programming
- Apply the various control structures and functions to real time problems
- Write Python functions to facilitate code reuse
- Make their code robust by handling errors and exceptions properly
- Ability to engage in independent and life-long learning in the broadest context of technological change.

*****

Second Year **BRIDGE COURSE VI** **Semester III**

INTERNET PROGRAMMING

Code: **(Theory)** **Credit: 4**

COURSE OBJECTIVES:

- To introduce .Net framework, HTML, development of Applets; to learn applet – servlet communication and JSP.

UNIT – I INTRODUCTION TO ASP.NET:

Architecture of .NET Framework – Life cycle of ASP.NET – Standard controls – Validation controls – Rich web controls – Data controls – Navigation controls.

UNIT – II HTML TAGS:

History of HTML – Structure of HTML – Basic Tags of HTML - List – Linking Document –Graphics to HTML Documents.

UNIT – III THE SERVLET LIFE CYCLE:

The Servlet Alternative – Servlet Reloading. Retrieving Information: Initialization Parameters – The Server – The Client – The Request.

UNIT – IV APPLET-SERVLET COMMUNICATION:

Communication Options - Daytime Server - Chat Server. Inter servlet Communication: Servlet Manipulation - Servlet Reuse - Servlet Collaboration.

UNIT – V INTRODUCING JAVA SERVER PAGES:

Java Server Pages –Use of JSP – The Web Programming Environment: Evolution of the Web Application – The Shift from Client-Side to Server-Side Solutions.

UNIT – VI CURRENT CONTOURS (For continuous internal assessment only):

Contemporary DevelopmentsRelatedtotheCourseduringtheSemesterConcerned.

REFERENCES:

1. Kogent (2010), ASP.NET 4.0 Black Book – Platinum Edition, DreamtechPress,New Delhi. (Unit – I)
2. Ivan Bayross, Web Enable Commercial Application Development using HTML, DHTML, Javascript, PERL CGI, BPB Publications, 2000. (Unit – II)
3. Java ™ Servlet Programming by Jason Hunter with William Crawford, O'Reilly Publishers, (Units – III, IV).
4. Phil Hanna, "JSP: The Complete Reference", McGraw-Hill, 2001. (Unit – V)
5. Mathew Mac Donald (2010), ASP.NET Complete Reference , Tata McGraw Hill publishing Company Ltd., New Delhi.

6. Thomas A. Powell, HTML and XHTML: The Complete Reference, Tata McGraw Hill, 4th Edition 2003.
7. Herbert Schildt, "JAVA 2 Complete References", TMH publications, 4th Edition, 2001.
8. https://www.tutorialspoint.com/jsp/index.htm
9. https://docs.microsoft.com/en-us/aspnet/tutorials

COURSE OUTCOMES:

- The students would have become familiar with the .Net framework, HTML, development of Applets; to learn applet – servlet communication and JSP.

*****

Second Year	**BRIDGE PRACTICAL III PYTHON PROGRAMMING**	**Semester III**
Code:	**(Practical)**	**Credit: 2**

Implement the following Concepts:

1. Flow controls, Functions and String Manipulation.
2. Operations on Tuples and Lists.
3. Operations on Sets and Dictionary.
4. Simple OOP – Constructors, Method Overloading, Inheritance.
5. Reading and Writing Files & Regular Expressions.

*****

Second Year **CORE COURSE X** **Semester IV**

CLOUD COMPUTING FUNDAMENTALS

Code: **(Theory)** **Credit: 5**

COURSE OBJECTIVES:

- To Introduce the Fundamentals of Cloud Computing.
- To be Familiar with the Cloud Architecture
- To Understand about Cloud Service Models and to Know the Concept of Virtualization in Cloud Computing

UNIT – I COMPUTING PARADIGMS:

High-performance computing, parallel computing, distributed computing, cluster computing, grid computing, cloud computing, bio-computing, mobile computing quantum computing, and optical computing .Nano-computing.

UNIT – II CLOUD COMPUTING FUNDAMENTALS:

Motivation for Cloud Computing, The Need for Cloud Computing, Defining Cloud Computing, Definition of Cloud computing, Cloud Computing Is a Service, Cloud Computing Is a Platform, Principles of Cloud computing, Five Essential Characteristics, Four Cloud Deployment Models.

UNIT – III CLOUD COMPUTING ARCHITECTURE AND MANAGEMENT:

Cloud architecture, Layer, Anatomy of the Cloud, Network Connectivity in Cloud Computing, Applications on the Cloud, Managing the Cloud, Managing the Cloud Infrastructure, Managing the Cloud application, Migrating, Application to Cloud, Phases of Cloud Migration Approaches for Cloud Migration.

UNIT – IV CLOUD SERVICE MODELS:

Infrastructure as a Service, Characteristics of IaaS, Suitability of IaaS, Pros and Cons of IaaS, Summary of IaaS Providers, Platform as a Service, Characteristics of PaaS, Suitability of PaaS, Pros and Cons of PaaS, Summary of PaaS Providers, Software as a Service, Characteristics of SaaS, Suitability of SaaS, Pros and Cons of SaaS, Summary of SaaS Providers. Other Cloud Service Models

UNIT – V CLOUD SERVICE PROVIDERS:

EMC, EMC IT, Captiva Cloud Toolkit, Google Cloud Platform, Cloud Storage, Google Cloud Connect, Google Cloud Print, Google App Engine, Amazon Web Services, Amazon Elastic Compute Cloud, Amazon Simple Storage Service, Amazon Simple Queue ,Service, Microsoft Windows Azure, Microsoft Assessment and Planning Toolkit, SharePoint, IBM Cloud Models, IBM Smart Cloud, SAP Labs, SAP HANA Cloud Platform, Virtualization Services Provided by SAP, Sales force, Sales Cloud, Service Cloud: Knowledge as a Service, Rack space, VMware, Manjra soft Aneka Platform

UNIT – VI CURRENT CONTOURS (For continuous internal assessment only):

Contemporary Developments Related to the Course during the Semester Concerned

REFERENCES:

1. Essentials of Cloud Computing :K. Chandrasekhran , CRC press, 2014
2. Cloud Computing: Principles and Paradigms by Rajkumar Buyya, James Broberg and Andrzej M. Goscinski, Wiley, 2011.
3. Distributed and Cloud Computing, Kai Hwang, Geoffery C. Fox, Jack J.Dongarra,Elsevier, 2012.
4. https://developer.ibm.com/components/cloud-ibm/tutorials/
5. https://www.javatpoint.com/cloud-computing-tutorial
6. https://www.w3schools.in/cloud-computing
7. http://index-of.co.uk/Cloud-Computing-Books/Essentials%20of%20cloud%20computing%20(2015).pdf
8. http://dhoto.lecturer.pens.ac.id/lecture_notes/internet_of_things/CLOUD%20COMPUTING%20Principles%20and%20Paradigms.pdf

COURSE OUTCOMES:

At the End of the Course the Student Should be able to

- Compare the Strengths and Limitations of Cloud Computing.
- Know about the Computing Paradigms
- Identify the Architecture, Infrastructure and Service Models in Cloud Computing.
- Understanding the Virtualization Concepts in the Cloud Environment.
- Understanding the Cloud Service Providers in the Market Today

*****

Second Year **CORE COURSE XI** **Semester IV**

MANAGERIAL SKILLS

Code: **(Theory)** **Credit: 5**

COURSE OBJECTIVES:

- To enable the students to learn the art of getting things done in the modern business world by learning topics like lateral thinking, decision making, balancing work and life
- To understand the corporate social responsibility, and work ethics.
- To Understand different types of Strategies

UNIT – I THINKING STRATEGIES:

Strategic thinking – meaning – questions- things included in Strategic thinking – Process consideration in Strategic thinking – Strategic thinking competencies – importance of Strategic thinking – characteristics of Strategic Thinkers – Points to be kept in mind in Strategic thinking.Lateral Thinking – meaning – why Lateral Thinking – when to use Lateral Thinking – Benefits of Lateral Thinking – Techniques used in Lateral Thinking – Who needs Lateral Thinking – How to use Lateral Thinking? – Conventional Vs Lateral Leaders – Questions asked by Lateral Leaders – becoming a Lateral leader

UNIT – II INTERPERSONAL STRATEGIES:

Conflict Resolution – meaning – points to be understood before studying conflict resolution – sources of conflict – common reactions to conflict – role of perception in conflict – steps for Conflict Resolution – Conflict handling matrix – Functional and Dysfunctional outcome of conflict. Negotiation skills – process – styles – outcome – principles involved – negotiation model – being a negotiator – qualities of a negotiator.

UNIT – III IMPLEMENTATION STRATEGIES:

Facing changes – meaning – characteristics –why changes –pace of changes – impact of resistance –Reasons for resistance – types of people in facing changes – introducing change. Facing challenges – meaning – importance – path to facing challenges – benefits of facing challenges.

UNIT – IV ACTION BASED STRATEGIES:

Risk taking - meaning – factors determining Risk Taking – Risk management – users of Risk Management – Steps in Risk Management. Effective decision making – meaning – approaches – methods – steps – Decision making at the work place.

UNIT – V BEHAVIOURAL STRATEGIES:

Motivation and Staying motivated – meaning – finding reason for being motivated – staying motivated at work place – staying motivated in negative work environment – staying motivated during crisis. Balancing work and life – meaning

– work satisfaction – gender differences – responsibility of the employers and employees – ways of balancing work and life – handling professional and personal demands – organizing your desk.

UNIT – VI CURRENT CONTOURS (For continuous internal assessment only):

Contemporary Developments Related to the Course during the Semester Concerned

REFERENCES:

1. Alex K. (2012) Soft Skills – Know Yourself & Know the World, S.Chand&Company LTD, Ram Nagar, New Delhi- 110 055. Mobile No :94425 14814 (Dr.K. Alex)
2. Meena.K and V.Ayothi (2013) A Book on Development of Soft Skills (Soft Skills: A Road Map to Success), P.R. Publishers & Distributors, No, B-20 & 21, V.M.M. Complex, Chatiram Bus Stand, Tiruchirappalli- 620 002.(Phone :0431-2702824: Mobile : 94433 70597, 98430 74472)
3. Emotional Quotient – Daniel Goleman
4. Power of the Plus factor – Norman Vincent Peale.
5. The Seven Habits of Highly Effective people – Stephen covey.
6. https://www.structural-learning.com/post/thinking-strategies-a-teachers-guide
7. https://www.tutorialspoint.com/management_concepts/basic_management_skills.htm

COURSE OUTCOMES:

- Understand the Thinking Strategies
- Get knowledge on Interpersonal Strategies
- Gain knowledge about implementation Strategies
- Learn the action based Strategies
- Understand the behavioural Strategies

*****

Second Year **PROJECT** **Semester-IV**

Code: **Credit: 11**

Each candidate shall be required to take up a Project Work and submit it at the end of the final year. The Head of the Department shall assign the Guide who, in turn, will suggest the Project Work to the student in the beginning of the final year. A copy of the Project Report will be submitted to the University through the Head of the Department on or before the date fixed by the University.

The Project will be evaluated by an internal and an external examiner nominated by the University. The candidate concerned will have to defend his/her Project through a Viva-voce.

ASSESSMENT /EVALUATION /VIVA-VOCE:

1. PROJECT REPORT EVALUATION (Both Internal & External):

I. Plan of the Project - 20 marks

II. Execution of the Plan/collection of Data / Organisation of Materials / Hypothesis, Testing etc and presentation of the report. - 45 marks

III. Individual initiative - 15 marks

2. VIVA-VOCE / INTERNAL& EXTERNAL - 20 marks

TOTAL **- 100 marks**

PASSING MINIMUM:

	Vivo-Voce 20 Marks	**Dissertation 80 Marks**
Project	40% out of 20 Marks (i.e. 8 Marks)	40% out of 80 marks (i.e. 32 marks)

A candidate shall be declared to have passed in the Project work if he/she gets not less than 40% in each of the Project Report and Viva-voce but not less than 50% in the aggregate of both the marks for Project Report and Viva-voce.

A candidate who gets less than 40% in the Project must resubmit the Project Report. Such candidates need to defend the resubmitted Project at the Viva-voce within a month. A maximum of 2 chances will be given to the candidate.

****