COURSE STRUCTURE

B.TECH.

COMPUTER SCIENCE & ENGINEERING

Specialization

in

Internet of Things (IoT)

Under

Choice Based Credit System (CBCS)

Credits Distributions

Sr. No.	Category	No. of Credits
1	Humanities and Social Sciences (HS)	25
2	Basic Sciences (BS)	19
3	Engineering Sciences (ES)	24
4	Professional Core (PC)	48/50
5	Professional Elective (PE)	31
6	Open Elective (OE)	16
7	Project Work (PW)	17
8	Mandatory Non Credit Courses (MNC) (4 Courses)	-
Total		**180/182**

First Semester

S. NO.	CODE	SUBJECT	TEACHING SCHEME			CREDITS	CONTACTS HRS/WK
			L	T	P		
1.	BMAS0101	Engineering Mathematics I	3	1	0	4	4
2.	BPHS0001	Engineering Physics	3	1	0	4	4
3.	BELH0001	English Language Skills for Communication – I	2	0	0	2	2
4.	BECG1001	Electronics Engineering	3	1	0	4	4
5.	BCSG0001	Python Programming	4	1	0	5	5
6.	BCSG0900	Introduction to IoT	4	0	0	4	5
PRACTICALS							
1.	BPHS0801	Engineering Physics Lab	0	0	2	1	2
2.	BELH0801	English Language Lab – I	0	0	2	1	2
3.	BECG0800	Electronics Lab I	0	0	2	1	2
4.	BMEG0801	Engineering Drawing Lab	0	0	2	1	2
5.	BCSG0800	Python Programming Lab	0	0	2	1	2
		TOTAL	**19**	**4**	**10**	**28**	**34**

Second Semester

S. NO.	CODE	SUBJECT	TEACHING SCHEME			CREDITS	CONTACTS HRS/WK
			L	T	P		
1.	BMAS0102	Engineering Mathematics II	3	1	0	4	4
2.	BEEG1001	Basic Electrical Engineering	3	1	0	4	4
3.	BELH0002	English Language Skills for Communication – II	2	0	0	2	2
4.	BCSG0901	IIOT Communication & Interface	4	0	0	4	5
5.	BMEG0001	Basic Mechanical Engineering	3	1	0	4	4
6.	BCSG2001	Computer Programming	3	0	0	3	3
PRACTICALS							
1.	BEEG0800	Electrical Engineering Lab	0	0	2	1	2
2.	BELH0802	English Language Lab – II	0	0	2	1	2
3.	BMEG0800	Engineering Workshop Practice Lab	0	0	2	1	2
4.	BCSC0800	Computer Programming Lab	0	0	2	1	2
		TOTAL	**18**	**3**	**8**	**25**	**30**

Program Core

S. NO.	CODE	SUBJECT	TEACHING SCHEME				CREDITS	CONTACTS HR/WK	PRE-REQUISITES
			L	T	P	J			
THEORY									
1.	BCSC0002	Object Oriented Programming	3	0	0	0	3	3	Programming
2.	BCSC0003	Database Management System	3	0	0	0	3	3	
3.	BCSC0004	Operating Systems	3	0	0	0	3	3	
4.	BCSC1005	Computer Organization	3	0	0	0	3	3	
5.	BCSC0006	Data Structures and Algorithms	3	1	0	0	4	4	Programming
6.	BCSC0007	Introduction to Microprocessors	3	0	0	0	3	3	Computer Organization
7.	BCSC0008	Computer Networks	3	1	0	0	4	4	
8.	BCSC0009	Software Engineering	3	0	0	0	3	3	
9.	BCSC1010	Discrete Mathematics	3	1	0	0	4	4	
10.	BCSC0011	Theory of Automata and Formal Language	3	1	0	0	4	4	
11.	BCSC0012	Design and Analysis of Algorithms	3	0	0	0	3	3	Programming, Data Structures
12.	BCSE0101	Digital Image Processing	3	0	0	0	3	3	Mathematics, Programming
PRACTICALS									
1.	BCSC0801	Object Oriented Programming Lab	0	0	2	0	1	2	Programming Lab
2,	BCSC0802	Database Management System Lab	0	0	2	0	1	2	
3.	BCSC0803	Operating Systems Lab	0	0	2	0	1	2	
4.	BCSC0804	Computer Organization Lab	0	0	2	0	1	2	
5.	BCSC0805	Data Structures and Algorithms Lab	0	0	2	0	1	2	Programming Lab
6.	BCSC0806	Microprocessors Lab	0	0	2	0	1	2	
7.	BCSC0807	Design and Analysis of Algorithms Lab	0	0	2	0	1	2	Programming, Data Structures
8.	BCSE0131	Digital Image Processing Lab	0	0	2	0	1	2	Programming

Program Elective

S. NO.	CODE	SUBJECT	TEACHING SCHEME				CREDITS	CONTACTS HR/WK	PRE-REQUISITES
			L	T	P	J			
LIST OF PROGRAMME ELECTIVES									
THEORY									
1.	BCSG0900	Introduction to IoT	4	0	0	0	4	5	
2.	BCSG0901	IIOT Communication & Interface	4	0	0	0	4	4	
3.	BCSE0902	Smart Industrial Connectivity	4	0	0	0	4	5	
4.	BCSE0903	Data Science And Analytics	4	0	0	0	4	4	
5.	BCSE0904	Machine Learning	4	0	0	0	4	4	
6.	BCSE0905	Artificial Intelligence	4	0	0	0	4	5	
7.	BCSE0202	Embedded System	3	0	0	0	3	3	
8.	BCSE0103	Soft Computing	3	0	0	0	3	3	
9.	BCSE0206	Parallel Algorithms	3	0	0	0	3	3	
10.	BCSE0208	Cloud Computing and Virtualization	3	0	0	0	3	3	
11.	BCSE0251	Full Stack Using Scripting Technologies	3	0	0	0	3	3	
12.	BCSE0252	Full Stack Using Node JS	3	0	0	0	3	3	
13.	BCSE0254	PHP - Scripting Language	3	0	0	0	3	3	
PRACTICALS									
1.	BCSE0234	Cloud Computing lab	0	0	2	0	1	2	
2.	BCSE0231	Embedded System Lab	0	0	2	0	1	2	
3.	BCSE0233	Parallel Algorithms Lab	0	0	2	0	1	2	
4.	BCSE0281	Full Stack Using Scripting Technologies Lab	0	0	2	0	1	2	
5.	BCSE0282	Full Stack Using Node JS Lab	0	0	2	0	1	2	
6.	BCSE0284	PHP - Scripting Language Lab	0	0	2	0	1	2	
7.	BCSE0132	Soft Computing Lab	0	0	2	0	1	2	

Projects

S. NO.	CODE	SUBJECT	TEACHING SCHEME				CREDITS	CONTACTS HR /WK	PRE- REQUISITES
			L	T	P	J			
1.	BCSJ0950	Mini Project – I	0	0	0	0	2	0	
2.	BCSJ0951	Mini Project – II	0	0	0	0	2	0	
3.	BCSJ0971	Project – Part I	0	0	0	0	3	0	
4.	BCSJ0972	Project – Part II	0	0	0	0	8	0	
5.	BCSJ0991	Industrial Training	0	0	0	0	2	0	
TOTAL			**0**	**0**	**0**	**0**	**17**	**0**	

Mandatory Non Graded Course

S. NO.	CODE	SUBJECT	TEACHING SCHEME				CREDITS	CONTACTS HR/WK	PRE- REQUISITES
			L	T	P	J			
THEORY									
1.	BCSM0001	Introduction to Cyber Security	2	0	0	0	0	2	
2.	BCHM0101	Disaster Management	2	0	0	0	0	2	
3.	MBAM0001	Basic Course in Entrepreneurship	2	0	0	0	0	2	
4.	MBAM0002	Leadership And Organizational Behavior	2	0	0	0	0	2	
5.	BCHM0202	Environmental Studies	2	0	0	0	2	2	
6.	BELM0001	Introduction to Bhagavad Gita	2	0	0	0	2	2	

Humanities and Social Sciences

S. NO.	CODE	SUBJECT	TEACHING SCHEME				CREDITS	CONTACTS HR/WK	PRE- REQUISITES
			L	T	P	J			
THEORY									
1.	BELH0001	English Language Skills for Communication – I	2	0	0	0	2	2	
2.	BELH0002	English Language Skills for Communication – II	2	0	0	0	2	2	
3.	BELH0003	English for Professional Purposes – I	2	0	0	0	2	2	
4.	BELH0004	English for Professional Purposes – II	2	0	0	0	2	2	
5.	BELH0006	Ethics & Values	2	0	0	0	2	2	
6.	MBAH0001	Industrial Management	3	0	0	0	3	3	
PRACTICALS									
1.	BELH0801	English Language Lab – I	0	0	2	0	1	2	
2.	BELH0802	English Language Lab – II	0	0	2	0	1	2	
3.	BTDH0301	Soft Skills – I	0	0	2	0	1	2	
4.	BTDH0302	Soft Skills – II	0	0	2	0	1	2	
5.	BTDH0303	Soft Skills – III	0	0	8	0	4	8	
6.	BTDH0304	Soft Skills – IV	0	0	8	0	4	8	
TOTAL			**13**	**0**	**24**	**0**	**25**	**37**	

Basic Sciences

S. NO.	CODE	SUBJECT	TEACHING SCHEME				CREDITS	CONTACT S HR/WK	PRE- REQUISITES
			L	T	P	J			
THEORY									
1.	BMAS0101	Engineering Mathematics I	3	1	0	0	4	4	
2.	BMAS0102	Engineering Mathematics II	3	1	0	0	4	4	
3.	BMAS1103	Engineering Mathematics III	3	1	0	0	4	4	
4.	BPHS0001	Engineering Physics	3	1	0	0	4	4	
5.	BCHS0201	Environmental Studies	2	0	0	0	2	2	
PRACTICALS									
1.	BPHS0801	Engineering Physics Lab	0	0	2	0	1	2	
TOTAL			**17**	**5**	**4**	**0**	**24**	**26**	

Engineering Sciences

S. NO.	CODE	SUBJECT	TEACHING SCHEME				CREDITS	CONTACTS HR/WK	PRE- REQUISITES
			L	T	P	J			
THEORY									
1.	BEEG0001	Basic Electrical Engineering	3	1	0	0	4	4	
2.	BECG0001	Electronics Engineering	3	1	0	0	4	4	
3.	BMEG0001	Basic Mechanical Engineering	3	1	0	0	4	4	
4.	BCSG1001	Python Programming	3	0	0	0	3	3	
5.	BCSG0002	Computer Programming	3	0	0	0	3	3	
PRACTICALS									
1.	BEEG0800	Electrical Engineering Lab	0	0	2	0	1	2	
2.	BECG0800	Electronics Lab I	0	0	2	0	1	2	
3.	BMEG0800	Engineering Workshop Practice Lab	0	0	2	0	1	2	
4.	BMEG0801	Engineering Drawing Lab	0	0	2	0	1	2	
5.	BCSG1800	Python Programming Lab	0	0	2	0	1	2	
6.	BCSG0801	Computer Programming Lab	0	0	2	0	1	2	
Total			**15**	**3**	**12**	**0**	**24**	**25**	

BCSG1001: PYTHON PROGRAMMING

Objective: *This course introduces the solving of mathematical problems using Python programming using Object Oriented concepts and its connectivity with database.*

Credits:05 **L-T-P-J:4-1-0-0**

Module No.	Content	Teaching Hours
I	Introduction to Python: Introduction and Basics; Setting up path Python Data Variables & Operators: Data Variables and its types, id () and type () functions, Coding Standards; **Control Structures:** if-else, elif, Nested if, Iteration Control structures, Break, Continue & Pass; **String Manipulation:** Accessing Strings, Basic Operations, String slices Function and Methods. **Lists:** Introduction, accessing list, Operations, Working with lists, Function and Methods. **Tuple:** Introduction, accessing tuples, Operations, Working, Functions and Methods. **Dictionaries:** Introduction, accessing values in dictionaries, Working with dictionaries, Properties, Functions.	22
II	**Functions:** Defining & Calling a function, Passing arguments to functions – Mutable & Immutable Data Types, Different types of arguments, Recursion, Scope of variables; **Modules and Packages:** User-defined modules and Standard Library: random, numpy, scipy, sys, Math Module, String Module, List Module, Date & Time Module, Regular Expressions: match, search, replace; **Input-Output:** Printing on screen, reading data from keyboard, Opening and closing file, Reading and writing files, Functions. **Exception Handling:** Exception, Exception Handling, except clause, try? finally clause, User Defined Exceptions. Basics of Python for Data Analysis, Introduction to series and data frames& Python using Pandas.	22

Text Books:

- Paul Barry: "Head First Python "O'Reilly Media, Inc.", 2010.

Reference Books:

- Bret Slatkin: "Effective Python: 59 Specific ways to write better Python", Addison Wesley, 2015.

Outcome: After completion of course, the student will be able to:

- CO1: Understand the basics of Python Programming.
- CO2: Apply the concepts of control structures and string manipulations of python programming.
- CO3: Understand the use of data structures available in Python List, Tuple and Dictionary.
- CO4: Experiment user-defined functions and access built-in functions.
- CO5: Experiment user-defined modules and access built-in modules- math, random, string, date, time, date time.
- CO6: Develop the programs using the concept of File Handling.
- CO7: Develop programs based on Exceptional Handling.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO2/PSO4
CO2	PO4/PSO1
CO3	PO5/PSO4
CO4	PO5,PO7/PSO1
CO5	PO2,PO8/PSO4
CO6	PO3,PO10/PSO2
CO7	PO5,PO9/PSO1

BCSG0002: COMPUTER PROGRAMMING

Objective: *To impart adequate knowledge on the need of problem solving techniques and develop programming skills to implements applications using the concepts of C Language. Also by learning the programming constructs they can easily switch over to any other language in future.*

Credits:05 **L-T-P-J:4-1-0-0**

Module No.	Content	Teaching Hours
I	**Generation of Programming Languages:** Low, Assembly, High and 4GL. **Language Processors:** Compiler, Interpreter, Assembler, Linker and Loader. **Algorithm:** Introduction, Features, Different Ways of stating Algorithms. **Flow Chart:** Introduction, Standard, Guidelines, Advantages and Limitations of using Flowcharts. **Basics of C:** Overview, Structure of a C program, Identifier, Keywords, Variables, Data types, Formatted Input and output. **Operators and Expression:** Assignment, Unary, Arithmetic, Relational, Logical, Bitwise, Conditional, Special operators and their precedence & Associativity. IEEE representation of data types like float & double, Lvalue and Rvalue **Type Conversion:** Type Promotion in expression, Conversion by Assignment, Truncation and Casting Arithmetic expression. **Decision and Case Control Structure:** if, if-else, nested if-else, Decisions using switch, switch versus if-else ladder, goto. **Loop Control Structure:** For loop, while loop, do-while loop, nesting of loops, break, and continue. **Arrays:** Introduction, one-dimensional and two-dimensional Array-Declaration, Initialization, Address Calculation. **Operations on Arrays:** Insertion, Deletion, Linear Search & Bubble Sort. **String:** Introduction, One dimensional and two dimensional Array-Declarations, Initialization **Operations on String:** Length, Copy, Reverse, Concatenate, Compare with & without built-in functions.	25
II	**Functions:** Declaration and Definition, Category of Functions, Parameter Passing Techniques – Call by Value, Passing Arrays to Functions. **Introduction to Storage Classes:** Auto, Static, Extern and Register. **Recursion:** Mechanics of Recursive Call, Implementation of Recursion, Recursion vs. Iteration. **The C Preprocessor:** Introduction, Macro Expansion and File Inclusion, Conditional Compilation and Miscellaneous Directives. **Pointers:** Declaration and Initialization of Pointer Variables, Accessing a Variable through its Pointer, Arrays and Pointers, Pointer and Strings, Pointer Arithmetic, Pointers to Pointers, Array of Pointers, Pointer to an Array, Two Dimensional Array and Pointers, Pointers to Functions, Dynamic Memory Allocation, void Pointer and Null Pointer. **User Defined Types:** enum, typedef, Union and Structure - Declaration, Initialization, Nested Structures, Arrays of Structures, Structure and Pointer, Passing Structure Through Function. Difference between Structures and Union. **File Handling:** Data and Information, File Concepts, File Organization, File Operations: Open, Read, and Close, Trouble in Opening a File. File Opening Modes, Working with Text Files. Random Access to Files of Records. Introduction to Command Line Arguments.	25

Text Books: Behrouz A. Forouzan and Richard F. Gilberg, "Computer Science – A Structured Programming Approach Using C", C Language Learning, 2007

Reference Books:

- Herbert Schildt, "C: The Complete Reference", 5th Edition, McGraw Hill Education
- K. N. King, "C Programming a Modern Approach", W. W. Norton, 2nd Edition, 2008.
- Kernighan and Ritche, "The C Programming Language", PHI, 2ndEdition, 2011.
- P. Dey and M. Ghosh, "Programming in C", Oxford University Press 2nd Edition, 2013.

Outcome: After completion of course, the student will be able to:

- CO1: Understand the basic concepts of problem solving skills.
- CO2: Apply the basic principles of programming in C language.
- CO3: Understand the concepts of arrays and strings in C language.
- CO4: Apply the concepts of functions to solve real world problems.
- CO5: Illustrate the concepts of recursion.
- CO6: Understand the concepts of pointers in C language.
- CO7: Understand the basic concepts of file handling.
- CO8: Develop algorithmic solutions to simple computational problems.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1,PO2,PO4,PO12/PSO1,PSO3
CO2	PO1,PO2,PO3,PO10/PSO1,PSO3
CO3	PO1,PO2,PO3,PO4/PSO1,PSO3
CO4	PO1,PO3, PO12/PSO1,PSO2
CO5	PO1,PO2,PO4 /PSO1,PSO3
CO6	PO1,PO2,PO3,PO4/PSO1,PSO2
CO7	PO1,PO3,PO6 /PSO1
CO8	PO1,PO2,PO4,PO10,PO12/PSO1,PSO3

BCSC0002: OBJECT ORIENTED PROGRAMMING

Objective: *This course introduces the Object-Oriented programming paradigm to students. It also teaches a student how to think objectively and model a Java program for solving real-world problems.*

CREDITS: 3 **L-T-P-J:3-0-0-0**

Module No.	Content	Teaching Hours
I	**Object-Oriented Programming:** Features of Object-Oriented Programming, Introduction to Object-Oriented Java Programming. **g Java Technology & Environment:** Understanding the compilation process of the JVM, JVM vs JDK vs JRE, Key Features of Java, Structure of a simple Java program. **Working with Java Primitive Data Types:** Strongly Typed nature of Java, Primitive Data Types in Java, The new 'var' keyword, Scope of a variable. **Accepting User Input in Java Programs:** using the Scanner class, using command line arguments. **Programming Constructs:** Sequence, Selection, Iteration & Transfer Statements, For-Each Loop. **Working with Java Arrays**: Declaring and Initializing One-Dimensional and Two-Dimensional Arrays in Java, Introduction to java. util. Arrays class. **The String API:** String Data Type, commonly used methods from the String API, String Tokenizer, String Builder & String Buffer. **Creating and Using Methods:** Signature of a method, Types of Methods, Overloading methods in a class, Static and Non-Static Methods. **Describing and Using Objects & Classes:** Declare the structure of a Java class, declaring members of a class (fields and methods), declaring and using Java Objects, lifecycle of an Object (creation, assignment, dereferencing and garbage collection), Constructors of a class, Overloading Constructors, Constructor chaining using 'this' and 'super' keyword. **Using Java Packages:** create and import Java packages and static imports, abstracting program logic to packages, creating executable main class, running the executable class inside a package. **Applying Encapsulation**: Using access modifiers with/in a class, principles of encapsulation. **Programming Abstractly Through Interfaces:** create and implement Interfaces for programs, private and default methods in Interfaces, declaring Abstract Classes, Constructors in Abstract Classes. Marker Interface, Functional Interfaces, Lambda Expressions in Java.	20
II	**Reusing Implementations using Inheritance:** Declaring Subclasses and Super classes, extend Abstract Classes, implementing Interfaces, exploring polymorphic behavior by overriding methods, Object Types vs Reference Types, differentiate overloading, overriding and hiding. **Exception Handling:** Exception Hierarchy, Need of Exception Handling, Checked Exceptions, Unchecked Exceptions and Errors, Try-Catch Blocks, Finally, Throw & Throws Keywords, creating and handling Custom Exceptions. **Threads in Java:** Life Cycle of a Thread, creating threads using Runnable and Thread, 'sleep ()', Thread Priorities. **Using Wrapper Classes:** Wrapper Classes in Java, Boxing-Unboxing-Auto Boxing-Auto Unboxing. **Generics & Collections:** Creating Generic classes, Generic Methods, Diamond Notation, Wildcards, Type Erasure, Collection Hierarchy, Base Interfaces, Lists, Sets and Maps. **The Stream API:** Introduction to the Stream API, using lambda expressions in Streams. **Regular Expressions:** Pattern and Matcher Class. **JDBC:** JDBC Drivers, Connecting to a MySQL Database, Driver Manager, Connection Interface, Statement Interface, Result Set Interface, Prepared Statements.	18

Text Book:

- Herbert Schildt, "The Complete Reference, Java Eleventh Edition", Oracle Press.2019.

Reference Book:

- Cay S Hosrtmann, "Core Java Volume I—Fundamentals, Eleventh Edition", Pearson,2018.
- Rogers Cadenhead, "Sams Teach Yourself Java in 21 Days (Covers Java 11/12), 8th Edition", Pearson,2020.

Outcomes: After completion of the course, students will be able to -

- CO1: Understand the basics of Object-Oriented Programming paradigm.
- CO2: Construct the logical flow of programs by using the sequence, selection, iterations and transfer statements.
- CO3: Apply the concepts of Object-Oriented Programming to model programs in Classes, Abstract Classes, Interfaces and Enums, and simplify program function by dissecting it into methods.
- CO4: Understand accessibility of members in a program unit and create packages to prevent namespace collisions.
- CO5: Predict run-time errors in a program by examining program functioning.
- CO6: Show the parallel processing capabilities of a program using a multithreading concept.
- CO7: Experiment with the predefined classes and interfaces defined in the Collections Framework.
- CO8: Develop a program using JDBC connectivity to demonstrate data persistence.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1,PO3/PSO1,PSO2
CO2	PO1,PO3/PSO1,PSO2
CO3	PO1,PO2/PSO1,PSO2
CO4	PO1/PSO2,PSO4
CO5	PO1,PO2,PO4/PSO4
CO6	PO1,PO2, PO3/PSO2
CO7	PO1,PO2,PO11/PSO2
CO8	PO1,PO2,PO3/PSO1,PSO2

BCSC0003: DATABASE MANAGEMENT SYSTEM

Objective: *The objective of the course is to enable students to understand and use a relational database& NoSQL system. Students learn how to design and create a good database.*

Credits: 03 **L-T-P-J: 3-0-0-0**

Module No.	Content	Teaching Hours
I	Introduction**:** An Overview of Database Management System, Database System Vs File System, Database System Concept and Architecture, Data Model Schema and Instances, Data Independence, Database Language and Interfaces (DDL, DML, DCL), Database Development Life Cycle (DDLC) with Case Studies. Data Modeling Using the Entity-Relationship Model**:** ER Model Concepts, Notation for ER Diagram, Mapping Constraints, Keys, Specialization, Generalization, Aggregation, Reduction of an ER Diagram to Tables, Extended ER Model. Relational Data Model and Language**:** Relational Data Model Concepts, Integrity Constraints, Entity Integrity, Referential Integrity, Keys Constraints, Domain Constraints, Relational Algebra **Database Design & Normalization I:** Functional Dependencies, Primary Key, Foreign Key, Candidate Key, Super Key, Normal Forms, First, Second, Third Normal Forms, BCNF, Non-Redundant Cover, Canonical Cover	20
II	**Database Design & Normalization II:** 4th Normal Form, 5th Normal Form, Lossless Join Decompositions, MVD and JDs, Inclusion Dependence. **File Organization:** Indexing, Structure of Index files and Types, Dense and Sparse Indexing **Transaction Processing Concept:** Transaction System, Testing of Serializability, Serializability of Schedules, Conflict & View Serializable Schedule, Recoverability, Recovery from Transaction Failures, Log Based Recovery, Deadlock Handling. **Concurrency Control Techniques:** Concurrency Control, Locking Techniques for Concurrency Control, 2PL, Time Stamping Protocols for Concurrency Control, Validation Based Protocol. **Distributed Database:** Introduction of Distributed Database, Data Fragmentation and Replication.	20

Text Books:

- Elmasri and Navathe, "Fundamentals of Database Systems", 6th Edition, Addison Wesley,2010.
- Sadalage, P. &Fowler, "NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence", Pearson Education,2012.

References Books:

- Date C J," An Introduction to Database Systems", 8th Edition, Addison Wesley.
- Korth, Silbertz and Sudarshan, "Database Concepts", 5th Edition, TMH,1998.
- Redmond, E. & Wilson, "Seven Databases in Seven Weeks: A Guide to Modern Databases and the NoSQL Movement", 1st Edition.

Outcome: After the completion of the course, the student will:

- CO1: Understand the concept of database management systems and Relational database.
- CO2: Identify the various data model used in database design.
- CO3: Design conceptual models of a database using ER modeling for real life applications and construct queries in Relational Algebra.
- CO4: Create and populate a RDBMS for a real life application, with constraints and keys using SQL.
- CO5: Select the information from a database by formulating complex queries in SQL.
- CO6: Analyze the existing design of a database schema and apply concepts of normalization to design an optimal database.
- CO7: Discuss indexing mechanisms for efficient retrieval of information from a database.
- CO8: Discuss recovery system and be familiar with introduction to web database, distributed databases.

Mapping of Course Outcomes (COs) withProgram Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1 /PSO1
CO2	PO2, PO3/ PSO2
CO3	PO2,PO3,PO6,PO11/PSO1,PSO2,PSO4
CO4	PO1,PO3/PSO1
CO5	PO1,PO5/PSO1
CO6	PO2,PO3,PO9/ PSO2
CO7	PO1,PO11 /PSO1
CO8	PO1,PO3,PO12/ PSO2

BCSC0004: OPERATING SYSTEMS

Objective: *This course aims to introducing the concept of computer organization. In particular, it focuses on basic hardware architectural issues that affect the nature and performance of software.*

Credits:03 **L-T-P-J:3-0-0-0**

Module No.	Content	Teaching Hours
I	**Introduction:** Operating System and its Classification - Batch, Interactive, Multiprogramming, Time sharing, Real Time System, Multiprocessor Systems, Multithreaded Systems, System Protection, System Calls, Reentrant Kernels, Operating System Structure- Layered structure, Monolithic and Microkernel Systems, Operating System Components, Operating System Functions and Services. **Processes:** Process Concept, Process States, Process State Transition Diagram, Process Control Block (PCB), Process Scheduling Concepts, Threads and their management. **CPU Scheduling:** Scheduling Concepts, Performance Criteria, Scheduling Algorithms, Multiprocessor Scheduling. **Process Synchronization:** Principle of Concurrency, Implementation of concurrency through fork/join and parbegin/parend, Inter Process Communication models and Schemes, Producer / Consumer Problem, Critical Section Problem, Dekker's solution, Peterson's solution, Semaphores, Synchronization Hardware. **Classical Problem in Concurrency:** Dining Philosopher Problem, Readers Writers Problem.	20
II	**Deadlock:** System model, Deadlock characterization, Prevention, Avoidance and detection, Recovery from deadlock, Combined Approach. **Memory Management:** Multiprogramming with fixed partitions, Multiprogramming with variable partitions, Paging, Segmentation, Paged segmentation. **Virtual memory concepts:** Demand paging, Performance of demand paging, Page replacement algorithms, Thrashing, Locality of reference. **I/O Management and Disk Scheduling:** I/O devices, I/O subsystems, I/O buffering, Disk storage and disk scheduling. **File System:** File concept, File organization and access mechanism, File directories, File allocation methods, Free space management.	20

Text Books:

- Silberschatz, Galvin and Gagne, "Operating Systems Concepts",9th Edition, Wiley, 2012.

Reference Books:

- SibsankarHalder and Alex a Aravind," Operating Systems", 6th Edition, Pearson Education, 2009.
- Harvey M Dietel, "An Introduction to Operating System", 2nd Edition, Pearson Education, 2002.
- D M Dhamdhere, "Operating Systems: A Concept Based Approach", 2nd Edition, 2006.
- M. J. Bach, "Design of the Unix Operating System", PHI, 1986.

Outcome: After completion of course, the student will be able to:

- CO1: Understand the classification of operating system environment.
- CO2: Understand the basic of process management.
- CO3: Apply the concept of CPU process scheduling for the given scenarios.
- CO4: Illustrate theprocess synchronization and concurrency process in operating system.
- CO5: Analyze the occurrence of deadlock in operating system.
- CO6: Describe and analyze the memory management and its allocation policies.
- CO7: Understand the concepts of disk scheduling.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1,PO2,PO7/PSO1
CO2	PO1,PO2 /PSO1
CO3	PO1,PO4/PSO1,POS3
CO4	PO3,PO4,PO6/PSO3,PSO4
CO5	PO1,PO4/PSO1,PSO3
CO6	PO1,PO2/PSO1,PSO3
CO7	PO1,PO2,PO7/PSO1,PSO3

BCSC1005: COMPUTER ORGANIZATION

Objective: *This course aims to introducing the concept of computer organization. In particular, it focuses on basic hardware architectural issues that affect the nature and performance of software.*

Credits: 03 **L–T–P-J: 3–0–0-0**

Module No.	Content	Teaching Hours
I	**Basic Organization:** Basic organization of the computer and Block level description of the functional units, Number representation; 1's and 2's Complement, Integer Representation, Arithmetic Addition & Subtraction with overflow. fixed and floating-point number representation, IEEE standard floating point representation. Introduction to Combinational Circuit- half adder, full adder, binary adder/subtractor, carry look ahead adders. Multiplexer and Demultiplexer, Register, bus and memory transfer. **Central Processing Unit:** Addition and subtraction of signed numbers, Multiplication: Signed operand multiplication, Booths algorithm. Processor organization, general registers organization, stack organization, Three, Two, One & Zero address instruction. Addressing modes, Micro-operations (Arithmetic, Logical & Shift) and its applications.	20
II	**Multiprogramming and Multiprocessing:** Flynn's classification, Introduction to pipelined operation. Instruction types, formats, Instruction cycles. **Control Unit:** Execution of a complete instruction. Hardwired and micro programmed control unit. Unconditional and Conditional branching. Microinstruction with next address field, pre-fetching microinstructions, Concept of horizontal and vertical microprogramming. **Memory:** Basic concept of Memory and its hierarchy, RAM memories, 2D, 2 & 1/2D memory organization. ROM memories. Cache memories: concept and design issues, performance, address mapping and replacement. Virtual memory: concept and implementation. **Input/Output:** Peripheral devices, I/O interface, I/O ports, Interrupts: interrupt hardware, types of interrupts and exceptions. Buses, bus architecture, types of buses and bus arbitration. Modes of Data Transfer: Programmed I/O, interrupt initiated I/O and Direct Memory Access., I/O channels and processors. Standard communication interfaces.	**20**

Text Books:

- M. Mano, "Computer System Architecture", 3rd Edition, PHI,1996

Reference Books:

- D.W. Patterson, "Computer Organization and Design", 4th **Edition**, Elsevier Publication, 2008.
- William Stalling, "Computer Organization",8th Edition, PHI, 2011.
- V. CarlHamacher, Zaky, "Computer Organization",4th International Edition, TMH, 1996.
- John P Hays, "Computer Organization", 2nd Edition, TMH.
- Tannenbaum , "Structured Computer Organization", 5thEdition, PHI, 2005.
- P Pal Chaudhry, "Computer Organization & Design", 2ndEdition, PHI, 2002.

Outcome: After completion of the course, the student will be able to:

- CO1: Understand the basics of digital computer system.
- CO2: Demonstrate the principle of arithmetic operations on unsigned, signed integers and floating point numbers.
- CO3: Understand the concepts of Combinational and Sequential circuits and their applications.
- CO4: Understand the CPU architecture and organization.
- CO5: Explain the basic concepts of pipelining.

- CO6: Design the steps for the execution of the complete instruction for hardwired and micro-programmed control unit.
- CO7: Explain the function of memory hierarchy.
- CO8: Determine the interface of CPU with input/output devices and their modes of transfer.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	**POs/PSOs**
CO1	PO1,PO3/PSO1
CO2	PO1,PO3/PSO1
CO3	PO2,PO3,PO5/PSO2
CO4	PO2,PO3,PO4/PSO1,PSO3
CO5	PO2,PO3,PO4/PSO2
CO6	PO1,PO2,PO3/PSO1,PSO3
CO7	PO2,PO3,PO5/PSO2,PSO3
CO8	PO3,PO4/PSO1

BCSC0006: DATA STRUCTURES AND ALGORITHMS

Objective: *The objective of this course is that students will construct and application of various data structures and abstract data types including lists, stacks, queues, trees and graphs.*

Credits: 04 **L–T–P-J: 3–1–0-0**

Module No.	Content	Teaching Hours
I	**Introduction:** Basic Terminology, Elementary Data Organization, Properties of an Algorithm, Efficiency of an Algorithm, Time and Space Complexity, Asymptotic Notations – Big-Oh; Operations on Data Structure, Abstract Data Types (ADT). **Linked Lists:** Implementation of Singly Linked Lists, Doubly Linked List, Circular Linked List, Operations on a Linked List - Insertion, Deletion, Traversal; Generalized Linked List, Polynomial Representation and Addition. **Stacks:** Primitive Stack Operations - Push & Pop, Array and Linked Implementation of Stack in C, Application of Stack: Prefix and Postfix Expressions, Evaluation of Postfix Expression, conversion of Infix to Postfix expression, Recursion, Principles of Recursion, Tail Recursion, Removal of Recursion, use of stack in Recursion, Tower of Hanoi Problem. **Queues:** Operations on Queue - Add, Delete operations, Implementation of Queue Using Array and Linked List, Circular Queues, Deque and Priority Queue. **Trees:** Basic Terminology, Array Representation and Dynamic Representation; Complete Binary Tree, Algebraic Expressions, Extended Binary Trees, Tree Traversal Algorithms - Inorder, Preorder and Postorder; Threaded Binary Trees, Traversing Threaded Binary Trees.	20
II	**Search Trees:** Binary Search Trees (BST), Insertion and Deletion in BST, AVL Trees, Introduction to M-Way Search Trees, B Trees. **Searching:** Sequential Search, Binary Search. **Sorting:** Bubble Sort, Selection Sort, Insertion Sort, Quick Sort, Two Way Merge Sort, and Heap Sort. **Graphs:** Terminology, Adjacency Matrices, Adjacency List, Graph Traversal - Depth First Search and Breadth First Search; Spanning Trees, Minimum Cost Spanning Trees – Prim's and Kruskal's Algorithm; Shortest Path Algorithm – Bellman-Ford and Dijkstra's Algorithm. **Hashing & Indexing:** Hash Function, Collision Resolution Strategies. Primary Indices, Secondary Indices, Indexing and Hashing Comparisons.	20

Text Book:

- Aaron M. Tanenbaum, YedidyahLangsam and Moshe J. Augenstein, "Data Structures Using C and C++", 2nd Edition, PHI, 2009.

Reference Books:

- Horowitz and Sahani, "Fundamentals of Data Structures", 3rd Edition, W H Freeman & Co, 2004-05.
- Jean Paul Trembley and Paul G. Sorenson, "An Introduction to Data Structures with Applications", 2nd Edition, TMH, 2007.
- R. Kruse, "Data Structures and Program Design in C", 2nd Edition, Pearson Education, 2004.
- Lipschutz Schaum's Outline Series, "Data Structures", 12th Reprint, TMH, 2010.
- G A V Pai, "Data Structures and Algorithms", TMH, 2009.

Outcome: After completion of course, student will be able to:

- CO1: Understand the basic concepts of the data structure and algorithms.
- CO2: Understand the complexity representation in terms of Big Oh, Theta and Omega notations.
- CO3: Apply the associated operations in linear data structure like stack, Queue and link list.
- CO4: Apply the associated operations in Binary Search Tree, AVL Tree and M- Way Search Tree.
- CO5: Understand the basic algorithms such as heap sort, graph traversal, quick sort, AVL trees, and hashing.
- CO6: Select the appropriate data structure to solve the problem.

- CO7: Apply the shortest path algorithm to solve real life problem.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	**POs/PSOs**
CO1	PO1/PSO1,PSO2
CO2	PO1, PO2/PSO1,PSO2
CO3	PO1/PSO1
CO4	PO1,PO4/PSO1
CO5	PO1,PO4/PSO3
CO6	PO2/PSO4
CO7	PO2/PSO4

BCSC0007: INTRODUCTION TO MICROPROCESSORS

Objective: *Objective of this subject is to introduce the basic concepts of microprocessor and assembly language programming. Identify and explain the operation of the components of typical microprocessor: the role of the ALU, registers, stack and the use of interrupts.*

Credits: 03 **L–T–P-J: 3–0–0-0**

Module No.	Content	Teaching Hours
I	**Introduction**: Microprocessors Evolution and Types, Basics of Pentium Microprocessor, Microprocessor Application, **8-Bit Microprocessor:**8085 Microprocessor and its Architecture, Addressing Modes, The 8085 Programming Model, Instruction Classification, Instruction Format, Overview of Instruction Set - Data Transfer Operation, Arithmetic Operation, Logic Operations and Branch Operations; Introduction to Assembly Language Program. **Programming Technique with Additional Instruction**: Looping, Counting, Indexing, Additional Data Transfer and 16-Bit Arithmetic Instruction, Counters and Time Delays, Stack and Subroutine.	20
II	**16 Bit Microprocessor:** Architecture of 8086 – Register Organization, Execution Unit, Bus Interface Unit, Signal Description, Physical Memory Organization, Mode of Operation, I/O Addressing Capabilities. **Peripheral Interfacing:** I/O Programming, Programmed I/O, Interrupt Driven I/O, DMA I/O, Memory-Mapped I/Os. **Peripheral Devices:** 8237 DMA Controller, 8255 Programmable Peripheral Interface, 8253/8254 Programmable Timer/Counter, 8259 Programmable Interrupt Controller.	18

Text Books:

- N Senthil Kumar, MSaravanan, and S Jeevananthan, "Microprocessors and Microcontrollers", Oxford University Press India, 2010.

Reference Books:

- Ramesh S. Gaonkar , "Microprocessor Architecture Programming and Applications with 8085", 4th Edition, Penram International Publishing, 2000.
- Ray A.K. Bhurchandi.K.M, "Advanced Microprocessor and Peripherals", TMH, 2002.
- D. V. Hall, "Microprocessors and Interfacing: Programming and Hardware", 2nd Edition, TMH, 1992.
- Y.C. Liu and G.A. Gibson, "Microcomputer Systems: The 8086/8088 Family Architecture Programming and Design", 2nd Edition, PHI, 2003.

Outcome: After the completion of the course, the student will be able to:

- CO1: Demonstrate the Microprocessor internal architecture and its operations.
- CO2: Develop programs based on 8085 microprocessor instruction set and addressing mode.
- CO3: Develop program using looping, counting, indexing, counter and time delays.
- CO4: Understand the concept of stack and subroutine for modular approach.
- CO5: Compare accepted standards and guidelines to select microprocessor (8085 & 8086) to meet performance requirements.
- CO6: Analyze the concept of interfacing the processor to external device with I/O programming & Interrupt Driven I/O.
- CO7: Understand the working of interfacing chips (8237, 8253/54, 8255 & 8259).

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1,PO2/PSO1
CO2	PO2,PO3/PSO1,PSO2
CO3	PO2,PO3/PSO1,PSO2
CO4	PO1,PO2,PO3/PSO1,PSO3
CO5	PO2,PO3,PO5/PSO1,PSO3
CO6	PO1,PO2/PSO3
CO7	PO1,PO2,PO4/PSO3

BCSC 0008: Computer Networks

Objective: *The objective is to understand fundamental underlying principles of computer networking, details and functionality of layered network architecture.*

Credits: 03 **Semester - IV** **L–T–P-J: 3–1–0-0**

Module No.	Content	Teaching Hours
I	**Introduction Concepts**: Goals and Applications of Networks, Network structure and architecture, The OSI reference model, services, Network Topology Design, Physical Layer Transmission Media, Line coding scheme, switching methods (circuit switching, Packet switching), TDM. **Medium Access sub layer**: Medium Access sub layer - Channel Allocations, LAN protocols - ALOHA protocols, CSMA, CSMA/CD, Overview of IEEE standards. **Data Link Layer**: Error detection and correction, Flow control (sliding window protocol)	20
II	**Network Layer**: Network Layer –IP addressing, subnet, CIDR, VLSM, Internetworking, Address mapping, routing. Connecting devices. **Transport Layer**: Transport Layer - Design issues, connection management, Flow control, TCP window management, congestion control-slow start algorithm. **Application Layer**: Data compression, Data Encryption, File Transfer, DNS, HTTP, SMTP, TELNET **Introduction to IPv6, transition from IPv4 to IPv6.**	20

Text Books:

- Forouzan B. A. , "Data Communication and Networking", 4th Edition, McGrawHill,2004.

References:

- Kurose, J.F. and Ross K.W., "Computer Networking: A Top-Down Approach Featuring the Internet", 3rd Edition, Addison-Wesley,2005.
- A.S. Tanenbaum, "Computer Networks", 2nd Edition, Prentice Hall India,2006.

Outcome: After the completion of the course, the student will be able to:

- CO1: Understand the concept of OSI and TCP/IP reference model.
- CO2: Understand the basics of data transmission at physical layer.
- CO3: Understand the channel allocation using ALOHA, CSMA and CSMA/CD.
- CO4: Apply error detection and correction technique to eliminate transmission error.
- CO5: Analyze the fixed and variable length address (IPv4) subneting for the given scenarios.
- CO6: Understand the design issues of the transport layer.
- CO7: Understand the mechanism of protocols at application layer such as FTP, HTTP, Telnet, DNS.
- CO8: Understand IPv6 addressing and differentiate it from IPv4.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1,PO3,PO12/PSO1
CO2	PO1/PSO2
CO3	PO1,PO4/PSO1,PSO4
CO4	PO1,PO3/PSO1
CO5	PO1,PO3,PO4,PO6/PSO3
CO6	PO2,PO4/PSO1
CO7	PO5,PO12/PSO2
CO8	PO4,PO7/PSO4

BCSC0009: SOFTWARE ENGINEERING

Objective: *Be employed in industry, government, or entrepreneurial endeavors to demonstrate professional advancement through significant technical achievements and expanded leadership responsibility.*

L–T–P-J: 3–0–0-0

Credits: 03

Module No.	Content	Teaching Hours
I	**Introductory Concepts**: The evolving role of software – characteristics, components and applications. **Process Models:** Waterfall Model, Prototyping, Incremental, Spiral. **Agile software Development**: Introduction to Agile, Agile software development framework. **Software Requirement Specification**: Requirement Process, SRS Components, Requirement Specifications with Use Cases Diagram. **Software Project Planning:** Project Planning Objectives. **Software Metrics:** Size, Function Point, Staffing, Project Estimation Methods–COCOMO Model. **Function-Oriented Design**: Problem Partitioning, Abstraction, Top Down and Bottom Up Design. **Module-Level Concepts:** Coupling, Cohesion, Design Notation and Specification - Structure Charts; Structured Design Methodology - Data Flow Diagram, Sequence Diagram.	20
II	**OO Analysis and OO Design:** OO Concepts, Introduction to UML Design Patterns: Class Diagram, Activity Diagram, State Chart Diagram. **Coding**: Coding Process, Verification – Code Inspections, Software Metrics. **Testing Fundamentals**: Test Case Design, Black Box Testing Strategies, White Box Testing, Unit Testing, Integration Testing, System Testing. **Introduction to Automation Testing and Testing Tools:** Automated Testing Process, Framework for Automation Testing, Introduction to Automation Testing Tool. **Software Quality:** Models, ISO 9000 Certification for Software Industry, SEI Capability Maturity Model. **Software Maintenance**: Models Cost of Maintenance, Re-engineering, Reverse Engineering.	18

Text Books:

- R. S. Pressman , "Software Engineering: A Practitioners Approach", 7thEdition, McGraw Hill,2010.

Reference Books:

- K. K. Aggarwal and Yogesh Singh , "Software Engineering", 3rd Edition, New Age International Publishers,2008.
- Rajib Mall , "Fundamentals of Software Engineering", 3rd Edition, PHI Publication,2009.
- R.E Fairley , "Software Engineering", McGraw Hill,2004.
- Sommerville, "Software Engineering", 9th Edition, Pearson Education,2010.

Outcome: After the completion of the course, the student will be able to:

- CO1: Understand the basic concepts of software engineering.

- CO2: Apply software processes to solve real world problems.
- CO3: Estimate the cost, effort and schedule of software using COCOMO Model.
- CO4: Analyze the software design techniques (structure chart, SDM, sequence diagram).
- CO5: Understand the basic concepts of OO analysis and design.
- CO6: Develop the test cases to validate the software.
- CO7: Understand the basic models of software Quality and maintenance.
-

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	**POs/PSOs**
CO1	PO1,PO7/PSO1
CO2	PO2,PO3/PSO4
CO3	PO2,PO11/PSO3
CO4	PO3,PO10/PSO4
CO5	PO3,PO7/PSO1
CO6	PO5,PO12/PSO2
CO7	PO4,PO9,PO12/PSO1

BCSC1010: DISCRETE MATHEMATICS

Objective: *The objective is to introduce students to language and methods of the area of Discrete Mathematics. The focus of the module is on basic mathematical concepts in discrete mathematics and on applications of discrete mathematics in computer science.*

Credits: 4 **L–T–P-J: 3–1–0-0**

Module No.	Content	Teaching Hours
I	**Sets, Relations and Functions:** Introduction to Set Theory, Venn diagrams, algebra of Sets, Inclusion-Exclusion Principle, Partitions, Relations, Properties and their types, Function and their types. Recurrence Relations and Generating Functions **Introduction to Counting Principle**: Permutation, Combination, Permutation with Repetition, Combination with Repetition, Pigeonhole Principle. **Posets & Lattices:** Definition, Properties of lattices – Bounded, Complemented, Modular and Complete lattice. **Natural Numbers:** Introduction, Mathematical Induction, Variants of Induction, Induction with Nonzero Base cases. Proof Methods, Proof by counter – example, Proof by contradiction.	20
II	**Propositional Logic** - Logical Connectives, Truth Tables, Normal Forms (Conjunctive and Disjunctive), Validity; **Predicate Logic** - Quantifiers, Inference Theory **Algebra**: Motivation of Algebraic Structures, Finite Groups, Subgroups and Group Homomorphism; Lagrange's Theorem; Commutative Rings and Elementary Properties; **Graph Theory:** Trees: Definition, Binary tree, Binary tree traversal, Binary search tree. Introduction to Graphs, , Operations on Graphs, Representation of graphs, Types: Planner, Directed, Complete, Bipartite Graph, Isomorphism, Euler Graph, Hamiltonian Graph, Connectivity.	20

Text Book:

- Kenneth H Rosen , "Discrete Mathematics and Its Applications", 7th edition, TMH,2012.

Reference Books:

- J.P. Tremblay , "Discrete Mathematical Structures with Applications to Computer Science", TMH, New Delhi,1997.
- V. Krishnamurthy , "Combinatorics: Theory and Applications", East-West Press, New Delhi,1986.
- Ralph P. Grimaldi, "Discrete and Combinatorial Mathematics- An Applied Introduction", 5th Edition, Pearson Education,2004.
- C.L. Liu, "Elements of Discrete Mathematics", 2nd Edition, TMH,2000.

Outcome: After the completion of the course, the student will be able to:

- CO1: Understand the notion of mathematical thinking and proofs to solve the problem.
- CO2: Apply the basics of discrete probability and number theory to solve the real world problem.
  CO3: Analyze basic discrete structures and algorithms using effectively algebraic techniques.
- CO4: Analyze mathematical concepts like sets, reasoning, relational algebra and graph theory to solve optimization problems.
- CO5. Analyze the validity of an argument using logical notation.
- CO6. Demonstrate the basic structures of proof techniques to write and evaluate the validity of arguments.
- CO7. Understand the basic principles of sets, set equalities and operations in sets.
- CO8. Apply counting principles to determine probabilities.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1,PO2/PSO1,PSO3
CO2	PO1,PO3/PSO4
CO3	PO2,PO3/PSO3
CO4	PO2,PO3/PSO3
CO5	PO1,PO2/ PSO3
CO6	PO1,PO3/PSO2,PSO3
CO7	PO1,PO2/PSO1
CO8	PO1,PO3/PSO1,PSO4

BCSC0011: THEORY OF AUTOMATA & FORMALLANGUAGES

Objective: *The objective of this course is that students will study and compare different models and views of the abstract notion of computation and its various aspects.*

Credits:04 **Semester V** **L-T-P-J:3-1-0-0**

Module No.	Content	Teaching Hours
I	**Introduction:** Alphabets, Strings and Languages; Automata and Grammars, Deterministic Finite Automata (DFA), Nondeterministic Finite Automata (NFA), Equivalence of NFA and DFA, Minimization of Finite Automata, Myhill-Nerode Theorem; FA with Output - Moore and Mealy machine, Applications and Limitations of FA. **Regular expression (RE):** Regular Expression to FA, DFA to Regular Expression, Arden Theorem, Non Regular Languages, Pumping Lemma for Regular Languages, Applications of Pumping Lemma, Closure Properties of Regular Languages. **Push Down Automata (PDA):** Introduction, Language of PDA, Acceptance by Final State, Acceptance by Empty Stack, Deterministic PDA.	20
II	**Context Free Grammar (CFG) and Context Free Languages (CFL):** Introduction, Derivation Trees, Ambiguity in Grammar, Ambiguous to Unambiguous CFG, Simplification of CFGs, Normal Forms for CFGs - CNF and GNF; Pumping lemma for CFLs, Equivalence of PDA and CFG. **Turing machines (TM):** Basic Model, Definition and Representation, Variants of Turing Machine and their equivalence, TM for Computing Integer Functions, Universal TM, Church's Thesis, Recursive and Recursively Enumerable Languages, Halting Problem, Introduction to Computational Complexity.	20

Text Books:

- K.L.P. Mishra and N. Chandrasekaran, "Theory of Computer Science: Automata, Languages and Computation", 3rd Edition, PHI,2006

Reference Books:

- Hopcroft, Ullman , "Introduction to Automata Theory, Languages and Computation", 3rd Edition, Pearson Education,2013.
- Martin J. C ," Introduction to Languages and Theory of Computations", 4th Edition, TMH,2011.

Outcome: After completion of course, the student will be able to:

- CO1: Understand the basic concepts of Context Free languages, Expression and Grammars.
- CO2: Analyze the conversion of NFA to DFA, Mealy to Moore and Moore to Mealy.
- CO3: Analyze the process to convert regular expression to DFA, DFA to regular expression, and minimization of DFA.
- CO4: Develop the PDA for the context free language and context free grammar.
- CO5: Analyze that the grammar is ambiguous or unambiguous.
- CO6: Apply the process to convert CFG to CNF and GNF.
- CO7: Understand the concept of Turing machine and its variants.
- CO8: Design the Turing machine for the real world application.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1/PSO1,PSO4
CO2	PO2,PO3/PSO3
CO3	PO2,PO3,PO9,PO12/PSO1,PSO3,PSO4
CO4	PO1,PO3,PO5,PO9/PSO3,PSO4
CO5	PO1,PO2,PO4/PSO3
CO6	PO2,PO3/PSO3
CO7	PO1,PO2/PSO1,PSO3
CO8	PO3,PO12/PSO1,PSO2,PSO3

BCSC0012: DESIGN & ANALYSIS OF ALGORITHMS

Objective: *The objective of this course is that students will construct and application of various data structures and concepts including Trees, Recursion & Dynamic programing.*

Credits:03 **L-T-P-J:3-0-0-0**

Module No.	Content	Teaching Hours
I	**Introduction:** Algorithms, analyzing algorithms, Complexity of algorithms, Growth of functions, Performance measurements, Sorting and order Statistics - Shell sort, Quick sort, Merge sort, Heap sort, Comparison of sorting algorithms, Sorting in linear time. **Advanced Data Structures:** Red-Black trees, B – trees, Binomial Heaps, Fibonacci Heaps. Divide and Conquer with examples such as Sorting, Matrix Multiplication, Convex hull and Searching.	20
II	Greedy methods with examples such as Optimal Reliability Allocation, Knapsack, Minimum Spanning trees – Prim's and Kruskal's algorithms, Single source shortest paths - Dijkstra's and Bellman Ford algorithms. Backtracking, Branch and Bound with examples such as Travelling Salesman Problem, Graph Coloring, n-Queen Problem, Hamiltonian Cycles and Sum of subsets Dynamic programming with examples such as Knapsack. All pair shortest paths – Warshal's and Floyd's algorithms, Resource allocation problem	20

Text Books:

- Thomas H. Coremen, Charles E. Leiserson and Ronald L. Rivest, Introduction to Algorithms, Third edition, Prentice Hall of India,2008.

Reference Books:

- Gilles Brassard Paul Bratley," Fundamentals of Algorithms", Prentice Hall,1996.
- Ellis Horowitz, SartajSahni, SanguthevarRajasekaran, "Fundamentals of Computer Algorithms", Orient Longman Pvt. Ltd,2008.
- Levitin , "An Introduction to Design and Analysis of Algorithms", Pearson,2008.

Outcome: After completion of course, student will be able to:

- CO1: Understanding of complexity representation in terms of Big Oh, Theta and Omega notations.
- CO2: Derive and solve recurrences describing the performance of divide-and-conquer algorithms (quick sort and merge sort).
- CO3: Compare and analyze different data structures (RB Tree, B Tree, Binomial Heaps, Fibonacci Heaps).
- CO4: Understand the major graph algorithms (DFS, BFS, Dijkstra's Bellman Ford) and their analyses.
- CO5: Understand the greedy paradigm and able to analyze when an algorithmic design situation calls for it. Synthesize greedy algorithms (Optimal Reliability Allocation, Minimum Spanning Trees, factorial Knapsack) and analyze them.
- CO6: Synthesize dynamic-programming algorithms (0/1 knapsack problem, Resource allocation problem, Warshal's and Floyd's algorithms) and analyze them.
- CO7: Understand the backtracking paradigm and able to analysis when an algorithmic design situation calls for it. Synthesize backtracking algorithms (N Queen Problem, TSP Problem, sum of subsets problem, Graph Coloring) and analyze them.
- CO8: Understand the branch and bound paradigm and able to analysis when an algorithmic design situation calls for it. Synthesize branch and bound algorithms (N Queen Problem, TSP Problem, Hamiltonian Cycles, Graph Coloring) and analyze them.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1,PO3,PO4,PO12/PSO1,PSO3
CO2	PO1,PO3,PO4,PO5/PSO1,PSO3
CO3	PO1,PO3,PO6/PSO1,PSO3
CO4	PO1,PO2,PO3,/PSO1,PSO3
CO5	PO1,PO2/PSO1,PSO3
CO6	PO1,PO2,PO3, PO6/PSO1,PSO3
CO7	PO1,,PO4,PO12/PSO1,PSO3
CO8	PO1,PO2,PO3,PO4,PO12/PSO1,PSO2

BCSE0101: DIGITAL IMAGE PROCESSING

Objective: *The objective is to introduce students the Fundamentals of digital Image processing. Students should study the basic of image operations and understand image analysis algorithm. Students can have exposure to current applications in the field of digital image processing*

Credits: 03 **L–T–P-J: 3–0–0-0**

Module No.	Content	Teaching Hours
I	**Introduction and Fundamentals:** Motivation and Perspective, Applications, Components of Image Processing System, Element of Visual Perception, A Simple Image Model, Sampling and Quantization, Some Basic Relationships between Pixels. **Intensity Transformations and Spatial Filtering:** Introduction, Some Basic Intensity Transformation Functions, Histogram Processing, Histogram Equalization, Histogram Specification, Local Enhancement, Enhancement using Arithmetic/Logic Operations – Image Subtraction, Image Averaging, Basics of Spatial Filtering, Smoothing - Mean Filter, Order Statistics Filters, Sharpening – The Laplacian. **Filtering in the Frequency Domain:** Fourier Transform and the Frequency Domain, Basis of Filtering in Frequency Domain	20
II	**Morphological Image Processing:** Introduction, Logical Operations involving Binary Images, Dilation and Erosion, Opening and Closing, The Hit-or-Miss Transformation, Morphological Algorithms – Boundary Extraction, Region Filling, Extraction of Connected Components, Convex Hull, Thinning, Thickening. **Image Segmentation:** Point, Line & Edge detection, Thresholding, Region-based Segmentation, Region Extraction - Pixel Based Approach & Region Based Approach, Edge and Line Detection - Basic Edge Detection, Canny Edge Detection, Edge Linking - Hough Transform. **Representation & Description:** Representation - Boundary Following, Chain Codes; Boundary Descriptors – Shape Numbers.	20

Text Books:

- R.C.Gonzalez and R.E.Woods, "Digital Image Processing", Prentice Hall, 3rd Edition,2011.

Reference Books:

- BhabatoshChanda and D. DuttaMajumder, "Digital Image Processing and Analysis", PHI,2011.
- S. Sridhar , "Digital Image Processing", Oxford University Press,2011

Outcome: After completion of course, student will be able to:

- CO1: Understand mathematical formulation of an image, its processing steps and relationship between image pixels.
- CO2: Apply Image enhancement using intensity transformations and spatial filtering.
- CO3: Analyze image enhancement for frequency domain using Fourier transform.
- CO4: Formulate region of interest through morphological operations.
- CO5:Evaluate strongly co-related regions obtained through Segmentation using discontinuity and homogeneity based segmentation techniques
- CO6: Describe an object of an image using Shape Number and Boundary descriptors.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1,PO2,PO3/PSO1,PSO3
CO2	PO3,PO5,PO11,PO12/PSO3,PSO4
CO3	PO1,PO2,PO3,PO7/PSO2,PSO3
CO4	PO1,PO2,PO4/PSO1,PSO2,PSO4
CO5	PO4,PO5,PO3/PSO1,PSO2,PSO3
CO6	PO9,PO10,PO11,PO12/PSO1,PSO2,PSO4

BCSC0014: APPLIED DATABASE MANAGEMENT SYSTEM

Objective: *The objective of the course is to enable students to understand and use a relational database& NoSQL system. Students learn how to design and create a good database.*

Credits:04 **L-T-P-J:4-0-0-0**

Module No.	Content	Teaching Hours
I	**Introduction:** An Overview of Database Management System, Database System vs File System, Database System Concept and Architecture, Data Model Schema and Instances, Data Independence, Database Language and Interfaces (DDL, DML, DCL), Database Development Life Cycle (DDLC) with case studies. **Data Modeling Using the Entity-Relationship Model:** ER Model Concepts, Notation for ER Diagram, Mapping Constraints, Keys, Specialization, Generalization, Aggregation, Reduction of an ER Diagram to Tables, Extended ER Model. **Relational Data Model and Language:** Relational Data Model Concepts, Integrity Constraints, Entity Integrity, Referential Integrity, Keys Constraints, Domain Constraints, Relational Algebra **Database Design & Normalization:** Functional Dependencies, Primary Key, Foreign Key, Candidate Key, Super Key, Normal Forms, First, Second, Third Normal Forms, BCNF, 4th Normal Form, 5th Normal Form, Lossless Join Decompositions, Non Redundant Cover, Canonical Cover, MVD and JDs, Inclusion Dependence.	26
II	**Transaction Processing Concept:** Transaction System, Testing of Serializability, Serializability of Schedules, Conflict & View Serializable Schedule, Recoverability, Recovery from Transaction Failures, Log Based Recovery, Deadlock Handling. **Concurrency Control Techniques:** Concurrency Control, Locking Techniques for Concurrency Control, 2PL, Time Stamping Protocols for Concurrency Control, Validation Based Protocol. **Distributed Database:** Introduction of Distributed Database, Data Fragmentation and Replication. **NoSQL System:** RDBMS vs NoSQL, BASE properties, Key-value, Columnar, Document and Graph-Based database, Introduction of MongoDB, Cassandra, Neo4j and Risk. **Database Programming using Python**: Database connectivity, Retrieving Data from Database, Parameters Passing, Execute many Methods, Cursor Attributes, Invoke Stored Procedures, Invoke Stored Functions.	26

Text Books:

- Elmasri and Navathe, "Fundamentals of Database Systems", 6th Edition, Addison Wesley, 2010.
- Sadalage, P. & Fowler , "NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence", Pearson Education, 2012.

References Books:

- Date C J," An Introduction to Database Systems", 8th Edition, Addison Wesley.
- Korth, Silbertz and Sudarshan, "Database Concepts", 5th Edition, TMH, 1998.
- Redmond, E. &Wilson, "Seven Databases in Seven Weeks: A Guide to Modern Databases and the NoSQL Movement", 1st Edition.

Course Outcome: After completion of course, student will be able to:

- CO1: Understand the concept of database management systems and Relational database.
- CO2: Identify the various data model used in database design.
- CO3: Design conceptual models of a database using ER modeling for real life applications and construct queries in Relational Algebra.

- CO4: Create and populate a RDBMS for a real life application, with constraints and keys, using SQL.
- CO5: Select the information from a database by formulating complex queries in SQL.
- CO6: Analyze the existing design of a database schema and apply concepts of normalization to design an optimal database.
- CO7: Discuss recovery system and be familiar with introduction to web database, distributed databases.
- CO8: Explain the differences between RDBMS and No-SQL,BASE properties and No-SQL databases.
- CO9:Design and implement the database system with the fundamental concepts of DBMS using Python.

Mapping of Course Outcomes (COs) withProgram Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	**POs/PSOs**
CO1	PO1/PSO1
CO2	PO2,PO3/PSO2
CO3	PO2,PO3,PO6,PO11/PSO1,PSO1,PSO2,PSO4
CO4	PO1,PO3/PSO1
CO5	PO1,PO5/PSO1
CO6	PO2,PO3/PSO2
CO7	PO1,PO3/PSO2
CO8	PO1,PO2,PO3/PSO1,PSO4
CO9	PO1,PO2,PO3,PO5/PSO1,PSO2,PSO4

BCSC0015: APPLIED DATA STRUCTURES AND ALGORITHMS

Objective: *The objective of this course is that students will construct and application of various data structures and abstract data types including lists, stacks, queues, trees and graphs.*

CREDITS: 05 **L-T-P-J: 4-1-0-0**

Module No.	Content	Teaching Hours
I	**Introduction**: Basic Terminologies, Elementary Data Organization with Arrays. **Algorithm**: Definition, Characteristics of an Algorithm, Time and Space Complexity. **Introduction to Asymptotic Notations**: Big-Oh, Big-Omega, Big-Theta **Operations on Data Structures**: Insertion, Deletion, Searching, Sorting, Traversing and Merging. **Abstract Data Types (ADT)**. **Linked Lists**: Implementation of Singly Linked Lists, Doubly Linked List, Circular Linked List, Operations on a Linked List - Insertion, Deletion, Traversal; Polynomial Representation and Addition. **Stacks**: Primitive Stack Operations - Push & Pop, Array and Linked List Implementation of Stack, Applications of Stack: Prefix and Postfix Expressions, Evaluation of Postfix Expression, conversion of Infix to Postfix expression. **Recursion**: Principles of Recursion, Head & Tail Recursion, Removal of Recursion, Use of stack in Recursion, Tower of Hanoi Problem, Nth Term of a Fibonacci Series. **Queues**: Operations on Queue – Enqueue & Dequeue operations, Implementation of Queue using Array and Linked List, Circular Queues, DEQueue. **Trees**: Basic Terminology, Array Representation and Linked Representation; Complete Binary Tree, Algebraic Expressions, Extended Binary Trees, Tree Traversal Algorithms - Inorder, Preorder and Postorder; **Search Trees**: Binary search trees, search efficiency, insertion and deletion operations, importance of balancing, AVL trees, searching, insertion and deletions in AVL trees, Tries, Red-Black Trees.	20
II	**Heaps**: Heaps as priority queues, heap implementation, insertion and deletion operations, binary heaps, binomial and Fibonacci heaps, heapsort, heaps in Huffman coding. **Hashing**: Search efficiency in lists and skip lists, hashing as a search structure, hash table, collision resolution. **Searching**: Sequential Search, Binary Search. **Sorting**: Bubble Sort, Selection Sort, Insertion Sort, Quick Sort, Merge Sort, Radix Sort, Bucket Sort. **Graphs**: Terminology, Adjacency Matrices, Adjacency List, Graph Traversal - Depth First Search and Breadth First Search; Spanning Trees, Minimum Cost Spanning Trees – Prim's and Kruskal's Algorithm; Shortest Path Algorithm – Bellman-Ford and Dijkstra's Algorithm. **Basic Algorithmic Techniques**: Greedy Algorithms, Divide & Conquer, Dynamic Programming. Backtracking. **Collections Framework:** Implementation of Collections Framework as Data Structures (Queue, Stack, List, Map, Set, Deque, PriorityQueue, Vector, Hashtable)	20

Text Books:

- Robert Lafore, "Data Structures And Algorithms in Java", 2nd Edition, Pearson SAMS,2003.

References Books:

- Elliot B. Koffman, Paul A. T. Wolfgang , "Data Structures: Abstraction and Design Using Java", 3rd Edition, Wiley,2016.

- Michael T. Goodrich, RobertoTamassia, Michael H. Goldwasser, "Data Structures and Algorithms in Java", 6th Edition, Wiley,2014.
- Horowitz and Sahani, "Fundamentals of Data Structures", 3rd Edition, W H Freeman & Co,2004-05.

Outcome: After completion of course, the student will be able to:

- CO1: Understand the basic concepts of the data structure and algorithms.
- CO2: Understand the complexity representation in terms of Big Oh, Theta and Omega notations.
- CO3: Apply the associated operations in linear data structure like stack, Queue and link list.
- CO4: Apply the associated operations in Binary Search Tree, AVL Tree and M- Way Search Tree.
- CO5: Understand the basic algorithms such as heap sort, graph traversal, quick sort, AVL trees, and hashing.
- CO6: Select the appropriate data structure to solve the problem.
- CO7: Apply the shortest path algorithm to solve real life problem.
- CO8: Understand the concepts of greedy approach, divide & conquer and dynamic programming.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1,PO2,PO4/PSO3
CO2	PO1,PO2,PO4/PSO3
CO3	PO1,PO2,PO3/PSO1,PSO4
CO4	PO1,PO2,PO3/PSO1,PSO2,PSO4
CO5	PO1,PO2,PO3,PO4,PO5/PSO1,PSO4
CO6	PO2,PO3/PSO1,PSO2
CO7	PO3,PO6,PO12/PSO1,PSO2
CO8	PO2,PO4/PSO3,PSO4

BCSG0800: PYTHON PROGRAMMING LAB

Objective: *This course introduces the solving of problems using Python programming using OO concepts and its connectivity with database.*

Credits:01 **L-T-P-J:0-0-2-0**

Module No.	Content	Lab Hours
I & II	Programs based on the concepts of: • Building Python Modules • Obtaining user Data • Printing desired output Programs based on the concepts of: • Conditional if statements • Nested if statements • Using else if and elif Programs based on the concepts of Iteration using different kinds of loops Usage of Data Structures • Strings • Lists • Tuples • Sets • Dictionary Program based on the concepts of User-defined modules and Standard Library (random, numpy, scipy, sys, Math Module, String Module, List Module). Program based on Input Output. Program based on exception Handling. Program based on Simple Data analysis. Program based on Pandas.	26

Text Books:

- Paul Barry: "Head First Python "O'Reilly Media, Inc.", 2010.

Reference Books:

- Bret Slatkin: "Effective Python: 59 Specific ways to write better Python", Addison Wesley, 2015.

Outcome: By the end of the course, students will learn to:

- CO1: Apply OO concepts using Python programming.
- CO2: Apply in-built packages defined in Python.
- CO3: Apply front-end as Python Programming to connect with any back-end.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO2/PSO1
CO2	PO3/PSO4
CO3	PO5/PSO2

BCSC0801: COMPUTER PROGRAMMING LAB

Objective: *The objective is to provide a comprehensive study of the C programming language. It stress the strengths of C, which provide students with the means of writing efficient, maintainable, and portable code.*

Credits:01 **L-T-P-J:0-0-2-0**

Module No.	Content	Lab Hours
I & II	• Mapping of flow chart, Algorithm, Language • Simple C-program execution • Programs based on various operators • Programs based on Decision and case Control Structure • Programs based on Loop Control Structure • Program based on special control statement ➢ break ➢ continue • Programs based on Array Insertion, Deletion, Linear Search & Bubble Sort • Programs based on String ➢Length, Copy, Reverse, Concatenate, Compare with & without built-in functions • Programs based on Functions. • Programs based on Storage Class. • Programs based on Recursion. • Programs based on Preprocessor. • Programs based on Pointers • Programs based on array • Programs based on string • Programs based on call by value and call by reference • Programs based on Dynamic Memory Allocation • Programs based on User Defined Data types ➢ Structure and Union ➢ Enum and Typedef • Programs based on File handling ➢ Opening a file ➢ Reading, writing and appending a file ➢ Closing file ➢ Random Access to Files of Records • Programs based on Command Line Argument.	52

Reference Books:

- Herbert Schildt, "C: The Complete Reference", 5th Edition, McGraw Hill Education
- K. N. King, "C Programming a Modern Approach", W. W. Norton, 2nd Edition, 2008.
- Kernighan and Ritche, "The C Programming Language", PHI, 2ndEdition, 2011.
- P. Dey and M. Ghosh, "Programming in C", Oxford University Press 2nd Edition, 2013.

Outcome: On Completion of this course, students are able to:

- CO1: Design programs involving decision structures, loops and functions.
- CO2: Understand the concepts of functions, recursion, pointers and file handling.
- CO3: Design programs involving structures, union and functions.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1,PO3/PSO1,PSO2
CO2	PO3,PO4/PSO1
CO3	PO3/PSO2,PSO4

BCSC0801: OBJECT ORIENTED PROGRAMMING LAB

Objective: *The objective of this course is that students will study and learn Object Oriented Modeling and programming.*

Credits:01 **L-T-P-J:0-0-2-0**

Module No.	Content	Teaching Hours
I& II	Programs in Java and python based on the concepts of: • Classes, Constructors, Polymorphism and Keyword Static. Programs based on the concepts of: • Inheritance, Multithreading Using Thread Class & Interface Runnable, String Handling, Generic Classes. Programs based on the concepts of: • Handling Database Connectivity. • Implementation of Collection Framework. Programs based on the concepts of: • Database Connectivity. • Retrieving Data from Database. • Parameters Passing, Execute many Method. • Cursor Attributes. • Invoke Stored Procedures. • Invoke Stored Functions.	24

Reference Books:

- Naughton, Schildt, "The Complete Reference JAVA2", 9th Edition, Oracle Press.
- Bhave&Patekar, "Programming with Java", Pearson Education
- Bret Slatkin: "Effective Python: 59 Specific ways to write better Python", Addison Wesley, 2015.

Outcome: After completion of course, the student will be able to:

- CO1: Implement object oriented language features.
- CO2: Design GUIs and Graphical programming.
- CO3: Design object oriented solutions for small systems involving database and event handling concepts.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1,PO2/PSO1
CO2	PO3,PO5/PSO2
CO3	PO3,PO5/PSO4

BCSC0802: DATABASE MANAGEMENT SYSTEM LAB

Objective: *The lab aims to develop an understanding of different applications and constructs of SQL, PL/SQL.*

Credits:01 **L-T-P-J:0-0-2-0**

Module No.	Content	Teaching Hours
I & II	• Write the SQL queries for data definition and data manipulation language. • To implement various operations on a table. • To implement various functions in SQL. • To implement restrictions on the table. • To implement the concept of the grouping of Data. • To implement the concept of Joins in SQL. • To implement the concept of sub-queries. • To implement the concept of views, sequence. • To implement the concept of PL/SQL using a cursor. • To implement the concept of Procedure function and Triggers.	24

References Books:

- Date C J," An Introduction to Database Systems", 8th Edition, Addison Wesley.
- Korth, Silbertz and Sudarshan, "Database Concepts", 5th Edition, TMH, 1998.
- Majumdar& Bhattacharya, "Database Management System", TMH

Outcome: After the completion of the course, the student will be able to:

- CO1: Apply SQL queries for DML and DDL.
- CO2: Develop the SQL queries for real life scenarios.
- CO3: Implement the procedural language (PL/SQL) and Triggers.

.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1,PO2/PSO1,PSO4
CO2	PO1,PO2/PSO1,PSO4
CO3	PO2,PO3,PO5/PSO2,PSO3

BCSC0803: OPERATING SYSTEMS LAB

Objective: *The lab aims to develop understanding the operation of UNIX operating system.*

Credits:01 **L-T-P-J:0-0-2-0**

Module No.	Content	Teaching Hours
I & II	• Implement the following basic commands (with options) used in UNIX/LINUX OS. • Write and implement the basic vi editor commands. • Shell scripts that use simple commands. • Decision based Shell scripts. • Shell scripts related to strings. • Shell scripts using pipes. • Shell scripts with loop statements. • Demonstration and solution for race condition. • Demonstration and use of System Calls. • Implement the basics of IPC in UNIX.	24

Reference Books:

- SibsankarHalder and Alex a Aravind," Operating Systems", 6th Edition, Pearson Education, 2009.
- Harvey M Dietel, "An Introduction to Operating System", 2nd Edition, Pearson Education, 2002.
- D M Dhamdhere, "Operating Systems: A Concept Based Approach", 2nd Edition, 2006.
- M. J. Bach. , "Design of the Unix Operating System", PHI, 1986.

Outcome: After completion of course, the student will be able to:

- CO1: Implement the basic operations on UNIX operating systems.
- CO2: Demonstrate the working of systems calls.
- CO3: Demonstrate message passing in Unix operating system.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1,PO3,PO4/PSO1
CO2	PO1,PO2/PSO1
CO3	PO1,PO4,PO5/PSO1,PSO2

BCSC0804: COMPUTER ORGANIZATION LAB

Objective: *The aim of the lab is to better understand the design of sequential Circuits such as Flip-Flops, Registers, and Counters.*

Credits: 01 **L–T–P-J: 0–0–2-0**

Module No.	Content	Lab Hours
I & II	• Bread Board Implementation of Flip-Flops. • Experiments with clocked Flip-Flops. • Design of Counters. • Bread Board implementation of Counters &Shift Registers. • Implementation of Arithmetic Algorithms. • Bread Board implementation of Adder/Subtraction (Half, Full). • Bread Board implementation of Binary Adder. • Bread Board implementation of Seven Segment Display. • Small Project based on combinational and sequential circuit.	24

Reference Books:

- D.W. Patterson , "Computer Organization and Design", 4th **Edition**, Elsevier Publication, 2008.
- William Stalling , "Computer Organization",8th Edition, PHI, 2011.
- M. Mano , "Computer System Architecture", 3rd Edition, PHI.

Outcome: After the completion of the course, the student will be able to:

- CO1: Implement the Combinational and Sequential Circuit.
- CO2: Demonstrate the working of counter and shift register.
- CO3: Demonstrate the working of ALU and seven segment displays.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO2,PO3,PO5/PSO2
CO2	PO3,PO4/PSO2
CO3	PO3,PO5/PSO1,PSO2

BCSC0805: DATA STRUCTURES & ALGORITHMS LAB

Objective: *The objective of this course is that students will understand and implement simple data structures, able demonstrate different sorting and searching techniques. and will be familiar with graphs and their applications.*

Credits:01 **L-T-P-J:0-0-2-0**

Module	Content	Lab
I &II	• Program to implement various operations in a singly linked list. • Program to implement insertion, deletion and traversal in a doubly linked List. • Program to implement polynomial addition using linked list. • Program to demonstrate the various operations on stack. • Program to convert an infix expression into postfix expression. • Program to evaluate a given postfix expression. • Program to implement Tower of Hanoi problem using Recursion. • Program to demonstrate the implementation of various operations on linear and circular queue. • Program to demonstrate the implementation of insertion and traversals on a binary search tree. • Program to implement Dijkstra's Algorithm to find the shortest path between source and destination. • Program to search a given element as entered by the user using sequential and binary search to search a given element as entered by the user. • Implementation of various sorting algorithms like Selection Sort, Bubble Sort, Insertion Sort, Merge Sort, Quick Sort and Heap Sort.	24

Note: All Code must be done in Java as well as Python

Outcome: After completion of course, student will be able to:

- CO1: Demonstrate the associated operations in linear data structure like stack, Queue and link list.
- CO2: Demonstrate the associated operations in Binary Search Tree and Dijkstra's Algorithm.
- CO3: Implementation the sorting algorithms like Selection Sort, Bubble Sort, Insertion Sort, Merge Sort, Quick Sort and Heap Sort.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1/PSO1
CO2	PO4/PSO1,PSO3
CO3	PO2/PSO3,PSO4

BCSC0806: MICROPROCESSORS LAB

Objective: *The objective is to introduce the Architecture and programming of the microprocessor and learning about interfacing and various applications of microprocessor.*

Credits: 01 **L–T–P-J: 0–0–2-0**

Module No.	Content	Lab Hours
I & II	• To study 8085 microprocessor System. • To study 8086 microprocessor System. • To develop and run basic programs in 8085 ALP. • To develop and run programs in 8085 ALP related to the concept of looping, counting and indexing. • To perform interfacing of RAM chip to 8085/8086. • To perform interfacing of keyboard controller. • To perform interfacing of DMA controller. • To perform interfacing of UART/USART.	24

Reference Books:

- Ramesh S. Gaonkar, "Microprocessor Architecture Programming and Applications with 8085", 4th Edition, Penram International Publishing, 2000.
- D. V. Hall , "Microprocessors and Interfacing: Programming and Hardware", 2nd Edition, TMH, 1992.

Outcome: After completion of course, student will be able to:

- CO1: Demonstrate the arithmetic and logical operations using assembly language programming (8085).
- CO2: Demonstrate the memory operations using assembly language programming (8085).
- CO3: Demonstrate the interfacing of Keyboard, DMA and UART controller.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1,PO3/PSO1,PSO2
CO2	PO1,PO2/PSO1,PSO2
CO3	PO1,PO3,PO5/ PSO2

BCSC0807: DESIGN & ANALYSIS OF ALGORITHMS LAB

***Objective*:** *The objective of this course is that students will understand and implement simple data structures, able demonstrate different sorting and searching techniques. and will be familiar with graphs and their applications.*

Credits:01 **L-T-P-J:0-0-2-0**

Module No.	Content	Teaching Hours

I& II	• Implementation of sorting algorithms:
	  ○ Insertion Sort
	  ○ Bubble Sort
	  ○ Selection Sort
	  ○ Divide and conquer approach:
	• Quick Sort
	• Merge Sort
	  ○ Heap Sort
	  ○ Counting Sort
	• Implementation of Searching Techniques:
	  ○ Linear Search
	  ○ Binary Search
	• Implementation of Matrix Multiplication
	• Implementation of Convex Hull
	• Implementation of Breadth First Search
	• Implementation of Depth First Search
	• Implementation of Greedy approaches:
	  ○ Optimal Reliability Allocation.
	  ○ Knapsack. Minimum
	  ○ Minimum Spanning trees:
	• Prim's and Kruskal's algorithms.
	  ○ Single source shortest paths –
	• Dijkstra's and Bellman Ford algorithms.
	• Implementation of Dynamic Programming:
	  ○ Longest Increasing Subsequence.
	  ○ Finding best path in maze.
	  ○ Matrix Chain Multiplication
	  ○ 0/1 Knapsack Problem
	  ○ Resource Allocation Problem
		32

Note: All Code must be done in Java as well as Python

Outcome: After completion of course, student will be able to:

- CO1: Implementation the sorting algorithms like Selection Sort, Bubble Sort, Insertion Sort, Merge Sort, Quick Sort and Heap Sort.
- CO2: Demonstrate and use the appropriate data structures for a given problem
- CO3: Implement the algorithms based on Greedy approach and Dynamic Programming.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1,PO2,PO4/PSO1,PSO2,PSO4
CO2	PO1,PO3,PO4/PSO1,PSO2,PSO3
CO3	PO2,PO3,PO5/PSO1,PSO2,PSO4

BCSE0131: DIGITAL IMAGE PROCESSING LAB

Objective: *The objective of this Lab is to develop hands-on experience to write programs using MATLAB/*Python *language for digital manipulation of images in both spatial and frequency domains.*

Credits: 01 **L–T–P-J: 0–0–2-0**

Module No.	Content	Lab Hours
I/II	• Basic commands to familiarize with MATLAB & perform the various Matrix operations. • Understanding image basic "image resize, image type conversion, extraction of color band, creating a synthesic image, pseudocolor image" • Perform various arithmetic operation (image addition, subtraction& complement) & logical operation (NOT, OR and XOR) on images • Perform various Image Enhancement operations: Image Negation function, Logarithmic Transformation, Power Law Transformation, Histogram Equalization, contrast stretching, plot histogram without using imhist function • Perform smoothing using linear (average filter) and order statistics filters (min, max & median) of varying sizes • Sharpen an image using Laplacian filter. • Perform various Fast Fourier transform (FFT) and frequency domain filtering on images using MATLAB. • Perform various Image Enhancement operation in frequency domain • Perform various Morphological operation dilation, erosion, internal & external boundary Extraction, Thinning,thickening of image &Perform Dilation, erosion, boundary Extraction without using direct function • Perform various thresholding segmentation (Simple, Multiple, and Adaptive thresholding) • Perform the various Edge Detection Operators (Ordinary, Roberts, Prewit, Sobeland Canny Operator) • Minor Application Assignment.	12*2=24

Reference Books:

- R.C. Gonzalez and R.E. Woods, "Digital Image Processing Using MATLAB", PHI, 2nd Edition,2010.
- Hands-On Image Processing with Python by SandipanDey, November 2018, Packt
- https://www.pyimagesearch.com/

Outcome: After studying the subject, the students will be able to:

- CO1: Implement digital image processing operations for image manipulation and Enhancement.
- CO2: Test the source code for Morphological and Segmentation algorithms.
- CO3: Apply image processing algorithms in real-life problems as Minor Application Assignment.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1,PO2,PO6,PO9/PSO1,PSO2
CO2	PO5,PO7,PO8,PO10/PSO1,PSO4
CO3	PO1,PO3,PO11,PO12/PSO3,PSO4

BCSC0808: APPLIED DATABASE MANAGEMENT SYSTEM LAB

Objective: *The lab aims to develop an understanding of different applications and constructs of SQL, PL/SQL and NoSQL databases.*

Credits:01 **L-T-P-J:0-0-2-0**

Module No.	Content	Teaching Hours
I & II	• Write the SQL queries for data definition and data manipulation language. • To implement various operations on a table. • To implement various functions in SQL. • To implement restrictions on the table. • To implement the concept of the grouping of Data. • To implement the concept of Joins in SQL. • To implement the concept of sub-queries. • To implement the concept of views, sequence. • To implement the concept of PL/SQL using a cursor. • To implement the concept of Procedure function and Triggers. • Introduction to MongoDB and its Installation on Windows or Linux, Description of mongo Shell, create database and show database, Commands for MongoDB and To study operations in MongoDB – Insert, Query, Update, Delete and Projection • To implement Database connectivity using Python	24

References Books:

- Date C J," An Introduction to Database Systems", 8th Edition, Addison Wesley.
- Korth, Silbertz and Sudarshan, "Database Concepts", 5th Edition, TMH, 1998.
- Majumdar& Bhattacharya, "Database Management System", TMH
- Sadalage, P. & Fowler , "NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence", Pearson Education, 2012.

Outcome: After the completion of the course, the student will be able to:

- CO1: Apply SQL queries for DML and DDL.
- CO2: Implement the procedural language (PL/SQL) and Triggers.
- CO3: Apply NoSQL queries in MongoDB.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1,PO2/PSO1,PSO4
CO2	PO2,PO3,PO5/PSO2,PSO3
CO3	PO5/PSO2

BCSC0809: APPLIED DATA STRUCTURES & ALGORITHMS LAB

Objective: *The objective of this course is that students will understand and implement simple data structures, able demonstrate different sorting and searching techniques. and will be familiar with graphs and their applications.*

Credits:01 **L-T-P-J:0-0-2-0**

Module No.	Content	Teaching Hours
I&II	• Program to implement various operations in a singly linked list. • Program to implement insertion, deletion and traversal in a doubly linked List. • Program to demonstrate the various operations on stack. • Program to implement Tower of Hanoi problem using Recursion. • Program to demonstrate the implementation of various operations on linear and a • circular queue. • Program to implement Dijkstra's Algorithm to find the shortest path between source and • destination. • Program to search a given element as entered by the user using binary search (divide and • conquer approach) to search a given element as entered by the user. • Implementation of various sorting algorithms like Selection Sort, Bubble Sort, Insertion Sort, Merge Sort, Quick Sort, Heap Sort, Radix Sort, Bucket Sort. • Program to implement AVL Trees. • Program to implement Red-Black Trees. • Program to implement Binary Heaps. • Program to implement Priority Queues. • Program to implement Huffman Character Length Encoding. • Program to implement tree traversal using Backtracking. • Project to create a LogBook to generate a Calendar Display. • Project to create a Hangman game using String Lexical Analysis.	24

Note: All Code must be done in Java as well as Python

Outcome: After completion of course, student will be able to:

- CO1: Demonstrate the associated operations in linear data structure like stack, Queue and link list.
- CO2: Demonstrate the associated operations in Binary Search Tree AVL Tree, Red- Black, and Dijkstra's Algorithm.
- CO3: Implementation the sorting algorithms like Selection Sort, Bubble Sort, Insertion Sort, Merge Sort, Quick Sort and Heap Sort.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1/PSO1
CO2	PO4/PSO1,PSO3
CO3	PO2/PSO3,PSO4

Program Elective (Only for Specialized Programme)

S. NO.	SEMESTER	SUBJECT	TEACHING SCHEME				CREDITS	CONTACTS HR/WK	PRE- REQUISITES
			L	T	P	J			
Bouquet: Internet of Things(IoT)									
THEORY									
1.	BCSG0900	Introduction to IoT	4	0	0	0	4	5	
2.	BCSG0901	IIOT Communication & Interface	4	0	0	0	4	4	
3.	BCSE0902	Smart Industrial Connectivity	4	0	0	0	4	5	
4.	BCSE0903	Data Science And Analytics	4	0	0	0	4	4	
5.	BCSE0904	Machine Learning	4	0	0	0	4	4	
6.	BCSE0905	Artificial Intelligence	4	0	0	0	4	5	
7.	BCSE0202	Embedded System	3	0	0	0	3	3	
8.	BCSE0103	Soft Computing	3	0	0	0	3	3	
9.	BCSE0206	Parallel Algorithms	3	0	0	0	3	3	
10.	BCSE0208	Cloud Computing and Virtualization	3	0	0	0	3	3	
11.	BCSE0251	Full Stack Using Scripting Technologies	3	0	0	0	3	3	
12.	BCSE0252	Full Stack Using Node JS	3	0	0	0	3	3	
13.	BCSE0254	PHP - Scripting Language	3	0	0	0	3	3	
PRACTICALS									
1.	BCSE0234	Cloud Computing lab	0	0	2	0	1	2	
2.	BCSE0231	Embedded System Lab	0	0	2	0	1	2	
3.	BCSE0233	Parallel Algorithms Lab	0	0	2	0	1	2	
4.	BCSE0281	Full Stack Using Scripting Technologies Lab	0	0	2	0	1	2	
5.	BCSE0282	Full Stack Using Node JS Lab	0	0	2	0	1	2	
6.	BCSE0284	PHP - Scripting Language Lab	0	0	2	0	1	2	
7.	BCSE0132	Soft Computing Lab	0	0	2	0	1	2	

BCSE0902: SMART INDUSTRIAL CONNECTIVITY

Objectives: *To learn designing of User Interface and Layouts for Android App, to learn how to use intents to broadcast data within and between Applications.*

Credits: 04 **L–T–P: 4–0–0**

Module No.	Contents	Teaching Hours (Approx.)
I	**Interfacing Raspberry Pi with Python:** Python Basic Programming, GPIO Pins Interfacing, Interfacing Sensors, **Thingworx:** Kapware Server, Connecting RPi to Thingworx, Real, Time patterns and anomaly detection, Integration tier & resource provider, Service tier, Thingworx apps. **ARM:** Embedded C and ARM Cortex Microcontrollers, Porting on, ARM Cortex Microcontrollers, Experience from building Osmosis platform, Vulnerabilities, Key aspects for Securing IoT solutions. **RFID:-**Introduction, Use Cases, Applications, interfacing with Arduino, Interfacing with RPi. **IoT Gateways:-**Introduction, Use Cases, Applications, interfacing with Arduino, Interfacing with RPi. **Bluetooth:-**Introduction, Use Cases, Applications, interfacing with Arduino, Interfacing with RPi. **ZigBee:-**Introduction, Use Cases, Applications, interfacing with Arduino, Interfacing with RPi, ZigBee Gateway Project.	25
II	**Raspberry Pi and Cloud Services:** Creating Local Server on RPi, Using RPi as a Gateway for IoT Devices, Cloud Architectures, Introduction to AWS, Deploying a Cloud Server, Setting Up a Cloud Server, Connecting RPi to Cloud, Sending and Receiving Data. **Data Management in IoT:-**Definition, IoT Data Lifecycle, IoT Data Management VS Traditional DBMS, AWS S3 Storage, AWS RDS Benefits of IoT Data Management. **Communication Channels:** GSM/GPRS, 2G, 3G, LTE, Wifi, PLC. **LPWAN:** LPWAN applications, LPWAN technologies, LoRA and LoRAWAN	25

Text Books:

- *DCS study material from portal,*
- *Internet Of Things (A Hands-on-Approach), By ArshdeepBahga, Vijay Madisetti.*

Reference Books: *DCS study material, Thingworx academics*

Outcome: After the completion of the course, the student will be able to:

- CO1: Demonstrate the installing of Android Studio and Cross Platform Integrated Development Environment.
- CO2: Design the User Interface and Layouts for Android App.
- CO3: Demonstrate the use intents to broadcast data within and between Applications.
- CO4: Understand the use of Content providers and Handle Databases using SQLite.
- CO5: Understand the basics of Android APIs for Camera and Music system.
- CO6: Explain various security issues with Android Platform.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1,PO2/PSO1
CO2	PO1,PO2,PO3/PSO1,PSO3
CO3	PO3,PO5/PSO1,PSO2
CO4	PO2,PO3/PSO4
CO5	PO1,PO3 /PSO1,PSO4
CO6	PO1,PO3/PSO2

BCSE0903: DATA SCIENCE AND ANALYTICS

Objectives: Conceptualization and summarization of big data and machine learning, trivial data versus big data, big data computing technologies, machine learning techniques, and scaling up machine learning approaches

Credits: 04 **L–T–P: 4–0–0**

Module No.	Contents	Teaching Hours
I	**Introduction to Data Analytics and Big Data:** An Overview Session for the Data Analyst, Data Scientist, Getting Started with Jupyter Notebook, Introduction to the Open Data Science Learning and, Competitive Platforms, IoT and Big Data, IoT Analytics Platform, Use Cases. **Scientific Computing with Python – Numpy:-** Introduction to List and Dictionary:-Basic operations in List and Dictionary, Importance of Numpy, Array Creation, Data Types, Unary Operations ,Shape Manipulation, Reshape, Transpose, Ravel, Universal Functions, Matrix Operations (Addition, Multiplication, Transpose and Inverse), Statistical Methods, Stacking (vstack and hstack), Splitting, Shallow copy and Deep copy/Cloning. **Data Analysis Workflow in Python using Pandas:** Introduction to Pandas, Pandas Data Structures, importing files/exporting files (*introduction to OS library), Series & Data Frame, Basic Functions on Data Frame, Indexing & Selecting Data, Selection by Level, Selection by Position, Boolean Selection, Sorting, Group by: Split-Apply- Combine, Handling Missing Data (Missing imputation), Data Analysis Scenarios. **Advanced Data Analysis using Pandas:** Merging of Data Frame, (Concat and Merge), Reshaping: Stack, Unstack, Pivot, Dummy/Indicator Variables, Working with Databases.	25
II	**Data Analysis Scenarios:** Converting Series to Time Series, Handling Invalid Data, Date-Time Index, Indexing, Time/Date Components, Period & Period Index, Parsing & Manipulating Dates. **Data Visualization:-**Introduction, Creating Different Types of Plots Scatter Plots, Line Graphs, Bar Plots, X and Y Ticks and Rotations ,Histograms, Box Plot, Stacked Plots. **Connect Azure IoT Hub to Thingworx:-**Install, configure, and run the ThingWorx Azure IoT Hub Connector, import devices that exist in Azure into ThingWorx, Connect a simulated Azure device to ThingWorx Foundation server. **Configure Permissions:** Configure and utilize the user access system, Control permissions at design time and run time **ThingWorx Developer Portal:** Developer Portal Home, ThingWorx Trials, Navigation: Resources, Navigation: Platform , Navigation: Apps, Navigation: AR, Navigation: Support	25

Text Books:

- *DCS study material from portal,*
- *"python data science handbook" by jakev anderpass*

Reference Books: *DCS study material*

Outcome: After the completion of the course, the student will be able to:

- CO1: Understand the overview of an exciting growing field of Big Data analytics.
- CO2: Analyze the Big Data using traditional data mining algorithms.
- CO3: Apply the tools required to manage and analyze big data.
- CO4: Understand the fundamental techniques and principles in achieving big data analytics with scalability and streaming capability.
- CO5: Understand the many types of big data like data streams.
- CO6: Solve complex real-world problems in for decision support.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1,PO3/PSO2
CO2	PO1,PO2,PO4/PSO1,PSO2
CO3	PO3,PO4/PSO1,PSO3
CO4	PO2,PO3/PSO4
CO5	PO2,PO3 /PSO1,PSO4
CO6	PO1,PO3/PSO2

BCSE0904: MACHINE LEARNING

Objectives: To provide Machine learning techniques: Three phases of machine learning, types of learning, support vector machine, decision trees and random forests, deep learning.

Credits: 04 **L–T–P: 4–0–0**

Module No.	Contents	Teaching Hours (Approx.)
I	**Techniques of Machine Learning:** Supervised learning, Unsupervised learning, Semi-supervised and Reinforcement learning, NLP. **Data Preprocessing**: Data Preprocessing, Data preparation, Feature engineering, Feature scaling, Dimensionality reduction. **Supervised learning**: Simple & Multiple Linear Regression, Interaction Terms, Nonlinear Transformations, Dummy variable regression, K-fold Cross Validation, Subset selection methods, Penalization [Ridge, Lasso, ElasticNet. **Introduction to Logistic Regression:** The Logistic Model cost function, Estimating the Coefficients, Making Predictions, Odds-Ratio, Performance Evaluation Matrices, [Sensitivity/Specificity/PPV/NPV, Precision, ROC curve etc.], Regularized Logistic Regression.	25
II	**Support Vector Machines**: Optimization Objective, The Maximal Margin Classifier, Kernel Method and Nonlinear Decision Boundaries One versus One Classification, One versus All Classification, Using Support Vector for Regression, Character recognition using SVM, Advantage and Disadvantages of SVM. Supervised learning:, Decision Tree:- CHAID, CART. **Supervised learning:** Random Forest: Motivating Random Forests: Decision Trees, Application of Random Forest, Ensembles of Estimators: Random Forests, Bagging and boosting, Example: Random Forest for Classifying Digits. Summary of Random Forests, Advantage and Disadvantages of RF. KNN (K-Nearest Neighbour):- Background of KNN, Application of KNN, Create a document retrieval system using k-nearest neighbors., Identify various similarity metrics for text data., Reduce computations in k-nearest neighbor search by using KD trees., Advantage and Disadvantages of KNN	25

Text Books:

- *DCS study material from portal,*
- *"python data science handbook" by jake vanderpass*

Reference Books: *DCS study material*

Outcome: After the completion of the course, the student will be able to:

- CO1: Apply the basic concepts of machine learning including bias-variance tradeoff.
- CO2: Apply the concepts of regression.
- CO3: Conceptualize supervise and re-enforcement learning for classification.
- CO4: Formulate the ensemble methods for improving classification.
- CO5: Apply ANN with optimization in machine learning.
- CO6: Design and develop projects based on machine learning.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1/PSO3
CO2	PO2,PO3/PSO1
CO3	PO4/PSO1,PSO3
CO4	PO2,PO3/PSO4
CO5	PO2,PO4/PSO4
CO6	PO1/PSO2,PSO4

BCSE0905: ARTIFICIAL INTELLIGENCE

Objectives: To create appreciation and understanding of both the achievements of AI and the theory underlying those achievements.

Credits: 04 **L–T–P: 4–0–0**

Module No.	Contents	Teaching Hours (Approx.)
I	**Supervised learning:** Introduction to Deep Learning: -Biological, neurons and artificial neurons, Non-linear Hypothesis, Model Representation, Examples & Intuitions, Transfer Function/ Activation Functions, Typical classes of network architectures, Feed forward ANN, Structures of Multi-layer feed forward networks, Back propagation algorithm, Back propagation. training and convergence, Local minima and global minima, Practical and design issues of back propagation learning, Advantage and Disadvantages. **Supervised learning: Ensemble and Stacking Model: -**Un -Supervised learning: Dimension Reduction Techniques: -Principal, Component Analysis (PCA), Eigen Value and Eigen Vectors,VARIMAX Rotation, Component Loading.	25
II	**Un - Supervised learning: Clustering**:-K means clustering, Fuzzy C means & K prototype, Application of Clustering algorithms Introducing different distance measure (Euclidean, Cosine,, Hamming etc), Measure of goodness of clusters o Davies-Bouldin Index, Dunn Index, Silhoutte coefficient, Advantages and Disadvantages of clustering algorithms. **Un - Supervised learning:** Recommendation Engine: -Collaborative filtering(CF), Item Based CF, User Based CF **NLP - Natural Language Processing:-**Introduction to Text mining and its importance, Text wrangling and cleansing, Tokenizing text and word net basics, Sentiment Analysis, Word cloud, Text Classification - Naive Bayesian. Text clustering, Topic mining - LDA (Latent Dirichlet allocation)	25

Text Books:

- *DCS study material from portal,*
- *"python data science handbook" by jake vanderpass*

Reference Books: *DCS study material*

Outcome: After the completion of the course, the student will be able to:

- CO1: Understand the basics of AI and the application areas.
- CO2: Apply the basic concepts of machine learning including bias-variance tradeoff, sample and true error.
- CO3: Explain the Supervised learning and un-supervised learning.
- CO4: Formulate the ensemble methods for improving classification.
- CO5: Design use of ANN with optimization in machine learning.
- CO6: Design and develop projects based on machine learning.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO2,PO3/PSO1,PSO3
CO2	PO2,PO3/ PSO3
CO3	PO2/PSO1,PSO3
CO4	PO2,PO3/PSO4
CO5	PO3,PO4/PSO2,PSO4
CO6	PO1,PO3/PSO2,PSO4

BCSE0906: EMBEDDED SYSTEM

Objective: *To acquaint students with methods of executive device control and to give them opportunity to apply and test those methods in practice.*

Credits: 03 **L–T–P-J: 3–0–0-0**

Module No.	Content	Teaching Hours
I.	**Introduction:** Introduction to Embedded Systems, Hardware Needs Challenges and Design Process of Embedded System, Processor Architectures, Memory Organization, Embedded Devices and Its Network. **Architecture of Embedded System**: CPUs, Bus Based Computer Systems, Programming Design and Analysis, Model of Program, Basic Compilation Techniques, Program Optimization and Performance. **OS for Embedded Systems:** Real Time Operating System, ISR in RTOS, Basic Design Using an RTOS, RTOS Task Scheduling Models, RTOS Programming.	20
II.	**Software Architectures:** Processes and Operating Systems, Multiple Task and Multiple Processes, Preemptive RTOS, Priority Based Scheduling, Inter-process Communication Mechanism, Multiprocessors. **Program Modeling Concepts:** Program Model, DFG Models, Modeling of Multiprocessor Systems, UML Modeling, Embedded Software Development Process and Tools. **Networks &System Design Techniques:** Networks for Embedded and its Design, Internet Enabled System, Introduction to Sensor Network, Design Methodologies, Requirement Analysis, System Analysis and Quality Assurance.	20

Text Books:

- Wolf, Wayne , "Computers as Components - Principles of Embedded Computing System Design", Elsevier,2008.

Reference Books:

- Raj Kamal , "Embedded Systems–Architecture, Programming & Design", Tata McGraw Hill,2011.
- David A. Simon , "An Embedded Software Primer", Pearson Education,1999.
- DanielW.Lewis, "Fundamentals of Embedded Software Where C and Assembly Meet", 2nd Edition, Pearson College Division,2012.
- James K. Peckol, "Embedded Systems: A Contemporary Design Tool", Wiley India,2012.

Outcome: After completion of course, student will be able to:

- CO1: understand the Process of Embedded System.
- CO2: Understand different Model of Program.
- CO3: Understand the concept behind the RTOS and its scheduling.
- CO4: Classify the software architecture into Multiple Task and Multiple Processes.
- CO5: Understand different Program Modeling Concepts.
- CO6: Apply knowledge of embedded systems along with some specialization in any area of computer engineering.

Mapping of Course Outcomes (COs) withProgram Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO4,PO5/PSO2
CO2	PO3 /PSO1
CO3	PO5/PSO2
CO4	PO4,PO5/PSO1
CO5	PO2,PO3/PSO2
CO6	PO3,PO10/PSO2

BCSE0907: PARALLEL ALGORITHM

Objective: *This course aims to introduce the concept of designing algorithms suitable for implementation on parallel computers. The focus will be on the algorithmic side. In particular, the emphasis will be on studying different parallel techniques and using them to design scalable parallel algorithms for a variety of problems and architectures.*

Credits: 03 **L–T–P-J: 3–0–0-0**

Module No.	Content	Teaching Hours
I	**Introduction:** Need for parallel computers, Models of computation, analyzing parallel algorithms, Expressing parallel algorithms **Algorithm Models:** Sequential model, need of alternative model, parallel computational models such as PRAM, LMCC, Hypercube, Cube Connected Cycle, Butterfly, Perfect Shuffle Computers, Tree model, Pyramid model,Fully Connected model, CRCW-CREW, EREW models **Performance Measures of Parallel Algorithms:** Speed-upandefficiency of PA, Cost- optimality, an example to illustrate Cost-optimal algorithms- such as summation, Min/Max on various models.	20
II	**Decomposition& Mapping techniques:** Database query processing, 15-puzzle problem, Parallel discrete event simulation. **Sorting:** Hyper quick sort, Bitonic merge sort, Enumeration sort (sorting on the CRCW model, CREW model and EREWmodel) **Searching and selection:** Searching on a sorted sequence (EREW, CREW, CRCW), Sequential selection algorithm, Parallel selection algorithm (EREW parallel solution) **Graph algorithms:** Graph coloring, Minimal spanning tree, Shortest path algorithm	20

Text Books:

- M.J. Quinn, "Designing Efficient Algorithms for Parallel Computer", McGraw-Hill, Inc, ISBN:0-07-051071-7,1987.

Reference Books:

- S.G. Akl, "Design and Analysis of Parallel Algorithms". Prentice-Hall, Inc., ISBN:0-13-200056-3,1989.
- S.G. Akl," Parallel Sorting Algorithm" by Academic Press,1985.
- AnanthGrama, Anshul Gupta, George Karypis, Vipin Kumar, "Introduction to Parallel Computing", Second Edition, Addison Wesley, ISBN: 0-201-64865,2003.

Outcome: After completion of course, student will be able to:

- CO1: understand the requirement for parallel computers.
- CO2: understand different Sequential and parallel computational models.
- CO3: Analyze the performance of Parallel Algorithms.
- CO4: Understand the system for parallel Database query processing.
- CO5: Understand the system for parallel searching and sorting algorithms.
- CO6: Understand the system for parallel graph algorithms.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	**POs/PSOs**
CO1	PO2,PO10/PSO2
CO2	PO5/PSO1
CO3	PO4/PSO2
CO4	PO1/PSO3
CO5	PO2 /PSO2
CO6	PO1,PO4/PSO3

BCSE0908: CLOUD COMPUTING

Objective: *This course covers aims to explain various technologies related to Cloud Computing and their practical implementations, discuss different architectural models of cloud computing, the concepts of virtualization and cloud orchestration.*

Credits: 03 **L–T–P-J: 3–0–1-0**

Module No.	Content	Teaching Hours
I	**Overview of Cloud Computing -** Brief history and Evolution of Cloud Computing, Traditional vs. Cloud Computing, Importance of Cloud Computing, Benefits and Challenges of Cloud Computing, Cloud computing vs. Cluster computing vs. Grid computing, Role of Open Standards Cloud **Computing Architecture:** Cloud computing stack Comparison with traditional computing architecture (client/server), Services provided at various levels, How Cloud Computing Works, Role of Networks in Cloud computing, protocols used, Role of Web services Service Models (XaaS) Infrastructure as a Service(IaaS), Platform as a Service(PaaS), Software as a Service(SaaS) Deployment Models Public cloud, Private cloud, Hybrid cloud, Community cloud. **Infrastructure as a Service(IaaS):** Introduction to virtualization, Different approaches to virtualization, Hypervisors, Machine Image, Virtual Machine(VM) Resource Virtualization Server, Storage, Network Virtual Machine (resource) provisioning and manageability, storage as a service, Data storage in cloud computing (storage as a service) Case Study: Amazon EC2. **Platform as a Service(PaaS):** Introduction to PaaS What is PaaS, Service Oriented Architecture (SOA) Cloud Platform and Management Computation Storage, Case study: Microsoft Azure as PaaS, Introduction, Service Offered, Creation of DB instance.	20
II	**Software as a Service (SaaS):** Introduction to SaaS, Web services, Web 2.0, Web OS, Open SaaS, SaaS with SOA Overview of Multi-Cloud Management Systems - Explain concept of multicloud management, Challenges in managing heterogeneous clouds, benefits of multi-cloud management systems. **Overview of Cloud Security -** Security concerns in Traditional IT, Challenges in Cloud Computing in terms of Application, Server, and Network Security. Security Concepts in VM, Abuse and Nefarious Use of Cloud Computing, Insecure Interfaces and APIs (Malicious Insiders, Shared Technology Issues, Data Loss or Leakage, Account or Service Hijacking, Unknown Risk Profile), Attacks in Cloud Computing **Cloud Security:** Infrastructure Security, Network level security, Host level security, Application level security Data security and Storage Data privacy and security Issues, Jurisdictional issues raised by Data Location Identity & Access Management, Access Control, Trust, Reputation, Risk, Authentication in cloud computing, IAM User, Groups and their Roles. **Service Management in Cloud Computing:** Service Level Agreements(SLAs), Billing & Accounting, Comparing Scaling Hardware: Traditional vs. Cloud, Economics of scaling: Benefitting enormously Managing Data Looking at Data, Scalability & Cloud Services Database & Data Stores in Cloud Large Scale Data Processing.	20

TextBooks:

- Raj Kumar Buyya, James Broberg, Andrezei M. Goscinski, "Cloud Computing": Principles and paradigms,2011.

ReferenceBook:

- Anthony T. Velte, Toby J. Velte, and Robert Elsenpeter Cloud Computing: A Practical Approach,2010.
- McGraw Hill. Rittinghouse, John, W, Cloud computing: Implementation, management and security.
- Barrie Sosinsky, Cloud Computing Bible, Wiley.2011.
- Rhoton, John, Cloud Computing Architected: Solution Design Handbook.
- Krutz, Ronald L.; Vines, Russell Dean, Cloud Security, A comprehensive Guide to Secure Cloud Computing.

Outcome: After successful completion of this student will be able to:

- CO1: Describe importance of virtualization along with their technologies like system, network, and storage virtualizations.
- CO2: Identify the architecture and infrastructure of cloud computing, including SaaS, PaaS, IaaS, XaaS, Public Cloud, Private Cloud, Hybrid Cloud and the core issues of cloud computing such as security, privacy, and interoperability.
- CO3: Justify the need of new technology of Virtualization & Cloud Computing and its ecological impact.
- CO4: Identify the known threats, risks, vulnerabilities and privacy issues associated with Cloud based IT services
- CO5: Apply fundamental concepts in cloud infrastructures to understand the tradeoffs in power, efficiency and cost
- CO6: Identify the Challenges in managing heterogeneous clouds.
- CO7: Analyze various cloud programming models and apply them to solve problems on the cloud.
- CO8: Describe the key components of Amazon web Service

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1,PO3,PO5,PO7/PSO2
CO2	PO1,PO3,PO7/PSO1
CO3	PO1,PO7/PSO1
CO4	PO1,PO3,PO5/PSO4
CO5	PO1,PO3,PO5,PO7/PSO4
CO6	PO1,PO3,PO5 /PSO2
CO7	PO1,PO3,PO5/PSO1
CO8	PO1,PO3,PO5,PO7/PSO1,PSO2

BCSE0909: CLOUD COMPUTING AND VIRTUALIZATION

Objective: *This course covers aims to explain various technologies related to Cloud Computing and their practical implementations, discuss different architectural models of cloud computing, the concepts of virtualization and cloud orchestration.*

Credits: 03 **L–T–P-J :3–0–0-0**

Module No.	Content	Teaching Hours
I	**Overview of Cloud Computing** - Brief history and Evolution of Cloud Computing, Traditional vs. Cloud Computing, Importance of Cloud Computing, Benefits and Challenges of Cloud Computing, Cloud computing vs. Cluster computing and Grid computing **Cloud Computing Architecture:** Cloud computing stack Comparison with traditional computing architecture (client/server), Services provided at various levels, How Cloud Computing Works, Role of Networks in Cloud computing, Role of Web services, Service Models (XaaS), Infrastructure as a Service(IaaS) , Platform as a Service(PaaS), Software as a Service(SaaS), Deployment Models, Public Cloud, Private Cloud, Hybrid Cloud, Community Cloud. **Infrastructure as a Service (IaaS)**: Introduction to virtualization, Different approaches to virtualization, Hypervisors and its types, Virtual Machine(VM), Resource Virtualization, Server Virtualization, Storage Virtualization, Network Virtualization, Virtual Machine Resource Provisioning and Manageability, Data storage in cloud computing (storage as a service), VM migration techniques, - Case Study: Amazon EC2. **Platform as a Service(PaaS):** Introduction to PaaS, Service Oriented Architecture (SOA), Cloud Platform and Management Computation Storage. **Software as a Service (SaaS:)** Introduction to SaaS, Web services, Web 2.0, Web OS, Open SaaS, SaaS with SOA.	19
II	**Load balancing**: Types of Load Balancing Scenarios in Cloud Computing Environment, Static, Dynamic and Centralized algorithms, Introduction to Open Stack, Horizon Web Interface, Using Keystone Identity Service, Swift Object Storage Service, Glance Image Service, Cinder Block Storage Service, Neutron Networking Service, and Nova Compute and Controller. **Overview of Cloud Security -** Security concerns in Traditional IT, Challenges in Cloud Computing in terms of Application, Server, and Network Security. Security Concepts in VM, Abuse and Nefarious Use of Cloud Computing, Insecure Interfaces and APIs (Malicious Insiders, Shared Technology Issues, Data Loss or Leakage, Account or Service Hijacking, Unknown Risk Profile), Attacks in Cloud Computing. **Cloud Security**: Infrastructure Security, Network level security, Host level security, Application level security Data security and Storage Data privacy and security Issues, **Energy Efficiency in Clouds**: Data Center Power Consumption, Green Data Centers. **Overview of Multi-Cloud Management Systems** - Explain concept of multi-cloud management, Challenges in managing heterogeneous clouds, benefits of multi-cloud management systems.	20

Text Book:

- Raj Kumar Buyya, James Broberg, Andrezei M. Goscinski, "Cloud Computing": Principles and paradigms,2011.

Reference Books:

- Anthony T. Velte, Toby J. Velte, and Robert Elsenpeter ," Cloud Computing": A Practical Approach, McGraw Hill,2010.
- Rittinghouse, John, W, "Cloud computing" : Implementation, management and security.

- Barrie Sosinsky," Cloud Computing Bible", Wiley. 2011
- Rhoton, John, "Cloud Computing Architected": Solution Design Handbook.
- Krutz, Ronald L.; Vines, Russell Dean, Cloud Security, A comprehensive Guide to Secure Cloud Computing.

Outcome: After completion of Lab, student will be able to:

- CO1: Identify the architecture and infrastructure of cloud computing, including SaaS, PaaS, IaaS, XaaS, Public Cloud, Private Cloud, Hybrid Cloud and the core issues of cloud computing such as security, privacy, and interoperability.
- CO2: Describe importance of virtualization along with their technologies like system, network, and storage virtualizations.
- CO3: Understand the concept of VM migration.
- Co4: Understand the concept of web services and SOA.
- CO5: Understand the concept of load balancing in cloud environment.
- CO6: Apply fundamental concepts in cloud infrastructures to understand the tradeoffs in power, efficiency and cost.
- CO7: Understand the concept of multi cloud management.
- CO8: Explain the key components of Amazon web Service and Microsoft Azure.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1,PO3,PO5,PO7/PSO1
CO2	PO1,PO3,PO5,PO7/PSO2
CO3	PO1,PO5/PSO1
CO4	PO1,PO3/PSO1
CO5	PO1,PO3,PO5/PSO1,PSO4
CO6	PO1,PO3,PO5,PO7/PSO4
CO7	PO1,PO3,PO5/PSO2
CO8	PO1,PO3,PO5,PO7/PSO1,PSO2

BCSE0910: FULL STACK USING NODE JS

Objective: *The objective is to provide a comprehensive study of the Backend. It stresses the strengths of Web (Full Stack), which provide students with the means of writing efficient, maintainable, and portable Website.*

Credits: 03 **L–T–P:J: 3–0–0–0**

Module No.	Content	Teaching Hours
I	**MongoDB:** Introduction to MongoDB, MongoDB Environment, MongoDB Create Database, MongoDB Drop Database, MongoDB Create Collection, MongoDB Drop Collection, MongoDB Read Operations, MongoDB Write Operations, MongoDB Data Modelling, MongoDB Administration. **Angular JS:** Introduction to AngularJS, AngularJS Expressions, AngularJS Modules, AngularJS Data Binding, AngularJS Scopes, AngularJS Directives & Events, AngularJS Controllers, AngularJS Filters, AngularJS Services, AngularJS HTTP, AngularJS Tables, AngularJS Select, Fetching Data from MySQL, AngularJS Validation, AngularJS API. **Express Framework:** Introduction to Express Framework, Introduction to Nodejs, what is Nodejs, Getting Started with Express, Express Routing,	20
II	**Express Framework:** Implementing MVC in Express, Middleware, Using Template Engines, Error Handling, API Handling, Debugging, Developing Template Engines, Using Process Managers, Security & Deployment. **Node.js:** Introduction to Node JS, Setup Dev Environment, Node Core, Node Modules, Creating Web server, File System, Debugging Node JS Application, Automation and Deployment, Events & Database connectivity. **React.js:** Welcome to Starting with React, React Components, React State and Props, React Event Handling, Routing in React React flux, &. Styling React	20

Text Books:

- Node.js, MongoDB and Angular Web Development: The definitive guide to using the MEAN stack to build web applications by Brad Dayley, Brendan Dayley, Caleb Dayley
- Express.js: Guide Book on Web framework for Node.js by Rick L.
- Introduction to React by Cory Gackenheimer,

Outcome: After completion of course, student will be able to:

- CO1: Apply programming concepts using Node.Js.
- CO2: Develop web application using MongoDB and AngularJs.
- CO3: Develop web application based on MongoDB.
- CO4: Understand project management and code.
- CO5: Develop RESTful and MVC based web application.
- CO6: Understand the basic concepts of React.js.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1/PSO1
CO2	PO3/PSO2
CO3	PO3/PSO3
CO4	PO2/PSO3
CO5	PO3/PSO2
CO6	PO2/PSO1

BCSE0981: FULL STACK USING NODE JSLAB

Objective: *The objective is to provide a comprehensive study of the Backend. It stresses the strengths of Web (Full Stack), which provide students with the means of writing efficient, maintainable, and portable Website.*

Credits: 01 **L–T–P-J: 0–0–2-0**

Module No.	Content	Lab Hours
I/II	• Installing and Managing MogoDB. • Create & Manage Database. • Create & Manage collections • Implementation of Modeling. • Create your first AngularJS application in Visual Studio. • Build a shopping cart using AngularJS • Implementation AngularJS Expressions • Implementation AngularJS Modules • Implementation AngularJS Events • Implementation AngularJS Filters & Services • Node JS- Setup Dev Environment. • Express Routing • Implementing MVC in Express. • Implementing Template Engines. • Implementing Node Modules, • Implementing React Components, • Implementing React Event.	12*2=24

Reference Books:

- Node.js, MongoDB and Angular Web Development: The definitive guide to using the MEAN stack to build web applications by Brad Dayley, Brendan Dayley, Caleb Dayley
- Express.js: Guide Book on Web framework for Node.js by Rick L.
- Introduction to React by Cory Gackenheimer,

Outcome: After studying the subject, the students will be able to:

- CO1: Implement web application using MongoDB and Angular.Js.
- CO2: Develop web application using NodeJs.
- CO3: Develop RESTful and MVC based web application.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO5/PSO2
CO2	PO3/PSO2
CO3	PO5/PSO4

BCSE0651: WIRELESS SENSOR NETWORKS (WST) & IOT STANDARDS

Objective:

Credits: **Semester III** **L–T–P-J: 3 – 3 –0-0**

Module No.	Content	Teaching Hours
I	**Introduction:** Introduction to Networks, Introduction to WSN, applications, topologies, Components of WSN – Transducer, Microcontroller, Battery, Transceiver, Geo-locator, Characteristics of WSN, WSN architecture, challenges of Deployment. **Energy:** Energy consumption, Energy efficiency, Energy scavenging. **OSI Network Model:** Introduction to different layers of OSI model. **WSN Simulation:** What is Simulation, types, Introduction to Simulator – QualNet,	20
II	**Routing:** Flooding based – Gossip, SPIN, Directed Diffusion, Hierarchical Based – LEACH, PEGASIS, TEEN, APTEEN, Location based Routing Protocols – MECN, SMECN, PRADA, QoS based – Speed, Minimum cost path routing. **Protocols:** Introduction to CoAP, MQTT, 6LoWPAN, Bluetooth, ZigBee, Z-WAVE, IPv6. **Operating Systems:** Introduction to WSN operating Systems – Tiny OS, Contiki, MANTIS. **Coverage and Localization –**Meaning of coverage and Localization**,** Full coverage, Perimeter Coverage. **IoT Open Source Tools:** Introduction to various IoT Open Source Tools.	20

Text Book:

- Wireless Sensor Networks (WST) and IoT Standards (IBM ICE Publication).

Reference Books:

- Fei Hu and Xiaojun Cao, "Wireless Sensor Networks Principles and Practice", CRC Press.
- Holger Karl and Andreas willig, "Protocol and Architecture for Wireless Sensor Networks", John Willey Publication.
- Feng zhao and Leonidas guibas, "Wireless Sensor Networks: an Information Processing Approach", Elsevier Publication.

Outcome: After completion of course, student will be able to:

- CO1: Understand the basics of WSN.
- CO2: Explain Energy Management mechanism in WSN.
- CO3: Develop the project using QualNet.
- CO4: Explain the routing protocols in WSN.
- CO5: Understand the WSN operating systems and IoT Open Source Tools.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1,PO12/PSO4
CO2	PO3/PSO1
CO3	PO5/PSO3
CO4	PO5,PO7/PSO2
CO5	PO4,PO5/PSO3

BCSE 0681: WIRELESS SENSOR NETWORKS (WST) & IOT LAB

OBJECTIVE: The objective is to enable students to work with different sensors and encourage them to think of different applications of WSNs.

Credits: **Semester III** **L–T–P-J: 0–0–2-0**

Module No.	Content	Lab Hours
I&II	• LED Blinking Experiment. • Using DHT sensor to measure Temperature and Humidity. • Using water level sensor to measure the level of water. • Experiment to turn on Buzzer automatically when temperature crosses a threshold value. • Experiment to turn on Buzzer automatically when water level crosses a threshold value. • Using Blynk to create a mobile application to turn on LED autiomatically. • Create different mobile applications using Blynk. • Access temperature data over the internet through ThingSpeak. • Creating different plots of temperature data on ThingSpeak. • Use distance sensor and send the distance information to ThingSpeak.	22

Outcome: After completion of course, student will be able to:

- CO1: Demonstrate the functions of different Sensors.
- CO2: Demonstrate the controlling of different sensors through mobile phone using Blynk.
- CO3: Demonstrate the accessing the sensor data over the internet through ThingSpeak.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO5,PO7/PSO4
CO2	PO10/PSO2
CO3	PO10/PSO2

BCSE0652: CLOUD ARCHITECTURE & DEPLOYMENT MODELS

Objective: The course enables students to understand the virtualization technology, Applications along with cloud computing concepts and services and to study different cloud architecture & deployment models.

Credits: 03 **Semester - IV** **L–T–P-J: 3–0–0-0**

Module No.	Content	Teaching Hours
I	**Introduction to Virtualization** Overview of Virtualization: Need of Virtualization, shortcoming of physical infrastructure, benefit of Virtualization, comparison of traditional IT infrastructure with virtualized infrastructure. **Virtualization** Implementing Virtualization, typical hardware / software server stack and its logical equivalence, pre/post virtualization server stack ,types of virtualization, area and technology based classification, history of virtualization, time sharing system, Extending Virtualization to x86 and its hardware support, impact of Virtualization: cost and manageability impact. **Server and Storage Virtualization** Types of Server Virtualization, simulation, Hardware Assisted Virtualization, Hypervisors, types of Hypervisors, Desktop Virtualization: Benefits Constraints and Types, storage Virtualization overview: benefit and types, features of logical layers, Host level storage Virtualization, host based mirroring, storage level Virtualization, network based storage Virtualization, Network and Application Virtualization. **Introduction to Cloud Computing** Overview: Introduction to cloud computing, Virtualization and cloud and its overlapping, types of services, business value, business impact of cloud, technological value of cloud, end user benefits, pros and cons of cloud model, anatomy of cloud, benefit of cloud, delivery and deployment model, different cloud architecture: public, private, hybrid and community its pros and cons, Service Models (XaaS), delivery models. Clint-server, cluster, grid models, cloud vs grid and their relationship, cluster and cloud, utility computing and evolution of cloud computing.	20
II	**Cloud computing Architecture:** Conceptual reference model, Cloud Computing solution components. Service Deployment, Cloud service management, IBM CC RA, SLA, Security and privacy **OpenStack:** Definition, Advantages, Releases, Architectural overview, Different components of Open Stack, Open stack- Hypervisors, Network Services, Storage - Block Storage, Object Storage, Choosing Storage Backends, Commodity Storage Backend Technologies: swift, Ceph, Gluster, Multiserver Openstack, Tenant model architecture, **Eucalyptus:** Introduction, Features and Functionality, Architecture, Basic and Advanced Components. Eucalyptus vs Openstack **OpenNebula:** Introduction, Features and Functionality, Architecture, Basic and Advanced Components. OpenNebula vs Openstack	20

Text Books:

- Introduction to Virtualization and Cloud Computing(IBM ICE Publication)
- Cloud Computing Architecture & Deployment Models (IBM ICE Publication)
- Raj Kumar Buyya, James Broberg, Andrezei M.Goscinski , Cloud Computing: Principles and paradigms, 2011.
- Bumgardner, V. C. , OpenStack in action. Manning Publications Company, 2016.

Outcome: After completion of course, student will be able to:

- CO1: Explain the core concepts of the cloud-computing paradigm.
- CO2: Describe the importance of virtualization along with their technologies like system, network, and storage virtualizations.
- CO3: Explain SaaS, PaaS, IaaS, XaaS, Public Cloud, Private Cloud and Hybrid Cloud.
- CO4: Describe the risk and security issues involved with the cloud computing Environment.
- CO5: Analyze the components of OpenStack
- CO6: Understand the Architecture and Components of Cloud Deployment and management tools like Eucalyptus, OpenNebula.

.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1/PSO1
CO2	PO1,PO3,PO5,PO7/PSO2
CO3	PO1,PO5/PSO1
CO4	PO1,PO3,PO5/PSO2
CO5	PO1,PO2/PSO1
CO6	PO1,PO2,PO5/PSO1

BCSE0682: CLOUD ARCHITECTURE AND DEPLOYMENT MODELS LAB

Objective: This course covers aims to explain various technologies related to Cloud Computing deployment models and their practical implementations, discuss different architectural models of cloud computing, the concepts of virtualization and cloud orchestration.

Credits: 01 **Semester IV** **L–T–P-J: 0–0–2-0**

Module No.	Content	Teaching Hours
I&II	1. a) Introduction to Packet Tracer. b) Network Topologies. (Including explanation of Simple PDU & Complex PDU.) 2. Connecting 3 netwoks using routers. Also, configure DHCP and DNS server. 3. Configuration of different Application services (SMTP, FTP, HTTP, TFTP, DHCP & DNS) 4. Configuration of Vlan and Inter- Vlan Routing. 5. Configure GRE over IP tunnel (VPN). 6. Static NAT configuration. 7. Configure Wireless network. 8. Configure different IoT devices. 9. Study on VMware a. Creating a VM b. Networking on VM c. Merging and splitting disk on VM d. Cloning the guest OS e. Deploying VM with template f. Creating Snapshots g. Managing Users, Groups, Permissions and Roles 10. Creating an EC2 instance on AWS 11. Configuration of db in AWS. 12. Creation of S3 bucket with single IAM user in AWS. 13. Deploying VM on Open Stack platform	18

Outcome: After completion of course, student will be able to:

- CO1: Implement the networking topologies and routing algorithms on Cisco Packet Tracer.
- CO2: Design Virtual Machines over Type- 1 & Type-2 Hypervisor & Test Client Server application over VMs created.
- CO3: Create the use cases of the key components of Amazon web Service.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1,PO3,PO5/PSO1
CO2	PO1,PO3,PO5/PSO2
CO3	PO1,PO3,PO5,PO7/PSO2

BCSE0653: DESCRIPTIVE ANALYTICS FOR IOT

Objective: *The purpose of the course is to provide an introduction to descriptive analytics in IoT. The concepts of business intelligence and its framework for IoT is discussed. The course provides an introduction to the concepts of big data and cloud in IoT. The various tools for IoT analytics is introduced with supporting case studies.*

Credits: 03 **Semester V** **L–T–P-J:3–0–0-0**

Module No.	Content	Teaching Hours
I	**Introduction to Analytics for IoT:** Definition, Internet of things and analytics, Types of analytics, Decision support systems: Business intelligence, Sample BI architecture, functional areas of BI tools, Collaborative BI **Descriptive analytics:** Introduction to descriptive analytics for IoT Benefits of BI, maximize value from BI systems, Strategy and business intelligence, five key areas of strategy, Planning a BI project. **BI framework for IoT:** BI architecture, Centralized versus decentralized architecture, System sizing, measurement and dependencies, authentication, authorization and access permissions, Server administration. BI framework for IoT. **Ecosystem preparation**: Device, network and software layer integrations, The functional architecture of IoT, Data mart framework, Data layer framework, Presentation layer framework.	20
II	**Data warehouse:** Characteristics, Multidimensional data, Classic star schema, Three-tier data warehouse architecture, Multi-tiered architecture Data analysis and OLAP **IoT data for analytics:** Data life cycle management, Sensor data: Problem of plenty data governance, Data collection: Capture of data, Data transportation: ETL strategy for IoT data, Data storage, Data processing: Analytics **Data consumption:** Visualization, Building data marts for IoT data OLAP: Server architectures, Typical OLAP operations, Dashboards and scorecards development, Dashboard types Layers of information Categorizing dashboards, Dashboard design principles, Dashboard design rules, Building reports, Data group and sort, Drilling in report: Drill up and down, Visualization of IoT data analytics **Big data analytics for IoT**: Introduction, Need for big data , Characteristics of big data, Structure of big data and need for standards, Big data analytics adoption, Benefits & barrier of big data analytics, Trends for big data analytics, Commoditization of hardware enabling new analytics, Wide ranging analyticsand techniques Big data platform and application frameworks, A big data platform manifestoBig data and IoT, BI from big data for IoT, context analytics, Real-time analytics. **Overview of cloud in big data:** Integration of devices on cloud for analytics, cloud and Big Data	20

Text Book:

- IBM Training Front cover Student Notebook Analytics for IoT

Reference Books:

Outcome: After completion of course, student will be able to:

- CO1: Understand the basics of analytics and types of analytics.
- CO2: Explain the BI architecture and BI components.
- CO3: Explain the different layer frameworks.
- CO4: Understand the basics of different tier data warehousing.
- CO5:Explain the OLAP operations.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1,PO2/PSO3
CO2	PO1,PO3/PSO2
CO3	PO1,PO3/PSO2
CO4	PO1,PO5/PSo2,PSO3
CO5	PO1,PO3/PSO3

BCSE0683: DESCRIPTIVE ANALYTICS FOR IOT LAB

Objective: *This course covers aims to explain various technologies related to Data Analytics with IoT for BI solutions, practical implementations of automated data scraping, screen recording etc, and discuss azure for virtualization and cloud orchestration.*

Credits: 01 **Semester V** **L–T–P-J: 0–0–2-0**

Module No.	Content	Teaching Hours
I&II	• Installation of UIPath • Process creation, Management of activity packages and configuration • Use of variables and Arguments • Use of sequence, flowcharts, and workflow • Perform input by using input methods • Perform reverse of string • Perform Automatic Recording with Desktop • Perform data scraping from Wikipedia • Perform Screen Scraping • Perform Image automation and OCR • Login to azure and subscribe student account • Raspberry configuration with azure IOT • Azure IOT Data Analysis with Power BI	20

Outcome: After completion of course, student will be able to:

- CO1: Implement the different automated Desktop Recording project.
- CO2: Design the Data Scraping from web.
- CO3: Understand Azure IOT Data Analysis with Power BI.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO1,PO3/PSO2
CO2	PO1,PO3/PSO2,PSO4
CO3	PO1,PO5/PSO2,PSO4

BCSE0654: IPv6 ANALYSIS AND APPLICATION

Objective: The course aims at creating knowledge on the networking concept especially in next generation IP along with its challenges, applications, routing and security. Student will also be achieving the learning in simulation of next generation IP.

Credits: 03 **Semester - IV** **L–T–P-J: 3–0–0-0**

Module No.	Content	Teaching Hours
I	**Introduction to IP:** Introduction to IP, IPv4, Limitations of IPv4, Exhaustion of IPv4 Addresses, Introduction to NAT. **IPv6:** Next generation IP (IPv6), Introduction and motivations behind IPv6 deployment, Workarounds In IPv6, current status and projections. **Dual stack, Tunneling, Challenges in deployment of IPv6, Structure of the IPv6 packet: base header,** Hop-by-Hop extension header, source routing and fragmentation. Comparing Ipv4 and IPv6 Header, IPv6 address architecture. Address categories and scopes. Current IPv6 prefix allocation, global, link-local, local unicast, loopback addresses. Mapping IPv4 addresses to IPv6 addresses, **Addresses in IPv6:** Multicast addresses, Anycast addresses, IPv6 Interface ID determination: privacy issues. Any cast addresses: use cases and possible implementations. Multicast addresses: scope definition and use cases. **Introduction to Changed Protocol and Routing:** RIPng, OSPFv3, BGPv4, DHCPv6, DNS, Stateless and Stateful auto-configuration, Introduction to QOS (Traffic class and Flow label). **Introduction to Packet tracer and Ipv4 and Ipv6 Simulation:** Overview & Simulation scenario.	20
II	**6LOWPAN and ICMPv6:** New functionalities; Neighbour Discovery protocol and address auto configuration. Interoperability with IPv4: dual-stack and tunnelling techniques. Introduction to 6LoWPAN and its architecture: simple, extended and ad-hoc networks. Issues in determining IPv6 links in LLNs and illustration of the undetermined link addressing model. IPv6 addressing in 6LoWPAN.6LoWPAN forwarding: route-over and mesh under approaches. Illustration of the 6LoWPAN adaptation layer header format. IP header compression: stateless vs. stateful options. IP header compression in 6LoWPAN. 6LoWPAN,ICMPv6. **Introduction to RPL:** The IPv6 Routing Protocol for LLNs, RPL: multi-point to point routing with destination oriented directed acyclic graphs. RPL instances, routing metrics and constraints, objective functions. RPL: loop detection and avoidance, P2MP and P2P routing CoAP proxying. Example of a CoAP-to-CoAP proxy. Resource observing and Service discovery. **Security Requirements on IPv6**	20

Text Books:

- Introduction to Ipv6 Analysis and Applications (IBM ICE Publication).
- Building the Internet of Things with IPv6 and MIPv6: The Evolving World of M2M Communications, John Wiley & Sons.

Reference Books:

- Hagen, S. . IPv6 essentials. Farnham: OReilly, 2002.

Course Outcomes:

- CO1: Understand the difference between IPv4 and IPv6.
- CO2: Demonstrate the mapping between IPv4 and IPv6.
- CO3: Explain the concepts of ICMPv6, 6LoWPAN and IPv6.
- CO4: Explain the basics of RPL.
- CO5: Demonstrate the simulations of IPv6 under Cisco Packet Tracer and Cooja simulator environment.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO10/PSO2
CO2	PO10/PSO1
CO3	PO1,PO10/PSO2
CO4	PO1,PO10/PSO1
CO5	PO5/PSO2

BCSC0655: BIG DATA ANALYTICS

Objective: *To cover the concept of big data and its analytics, Hadoop map reduce, mongo database and cassendra file system*

Credits: 04 **Semester II** **L–T–P-J: 3–0–0-0**

Module No.	Content	Teaching Hours
I	**Big Data:** Data-Definition, Representation, Types of Digital Data: Structural, Semi-Structural & Unstructured, Need of Big Data Big Data – Definition, Characteristics of Big Data, Units to measure Big data, Big data types, Benefits & Barrier of Big Data Analytics, Big data Sources, Scalability and Scaling up techniques. - Concept of Horizontal and Vertical scalability. **Big data computing environment**: Hadoop Fundamentals: What is Hadoop? Introduction to Hadoop, Data Distribution, HDFS-Hadoop Distributed File System, Name Nodes, Data nodes, DataNodes with Blocks of Multiple Files with a Replica. Yarn, Hadoop 1.0, Hadoop 2.0 and Hadoop3.0 **MapReduce:** an example of MapReduce, The Map function, Sort phase, Reduce function, Combiner and Partition functions, MapReduce example: wordcount. NoSQL technology, CAP Theorem, ACID Properties. **HADOOP Ecosystem and Flume**: Introduction to Hadoop Ecosystem, Introdcution to Sqoop, Zookeeper.	21
II	**Introduction to Mongo DB**: RDBMS vs. MongoDB, JSON, Unique Key, Dynamic Queries, Sharding, Replication, MongoDB QL: Create, Drop Database and Collections, CRUD: Create, Insert, Find, Update, Delete, Map Reduce Programming, Aggregations **Introduction to Cassandra DB**: Features of Cassandra, CQL Data Types, CQLSH: CRUD, Counter, TTL, List, Set, Map, Tracing, Import Export csv files **Introduction to HBase**: What is HBase? HBase Architecture, HBase Components, Data model, HBase Storage Hierarchy, Cross-Datacenter Replication, Auto Sharding and Distribution, Bloom Filter and Fold, Store, and Shift **Introduction to HIVE**: Hive Architecture, Hive Data types, Hive Collection Types, Hive File Formats, Hive Query Language, Hive Partitions, Bucketing, Views. **Introduction to Pig**: History and Anatomy of Pig, Pig on Hadoop, Use Case for Pig, Pig Primitive Data Types, Pig Latin Overview, Execution Modes of Pig, Field, Tuple, Bag, User Defined Function, Parameters in Pig, Piggy Bank, Word count example using Pig, Pig vs Hive.	20

References:

- Seema Acharya, Subhashini Chellappan , "Big Data and Analytics", First Edition, WILEY, 2015.
- Chuck Lam , "Hadoop in Action", Second Edition, Manning, 2018.

Outcomes:

- PO1: Understand Architecture for Big Data.
- PO2: Understand concept of Hadoop and its various versions.
- PO3: Understands YARN, HDFS and Map Reduce Algorithm.
- PO4: Understand Hadoop Eco system.
- PO5: Understand data access through HIVE, PIG etc.
- PO6: Understand MongoDB database and Cassandra file system.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO10/PSO2
CO2	PO10/PSO1
CO3	PO1,PO10/PSO2
CO4	PO1,PO10/PSO1
CO5	PO5/PSO2
CO6	PO10/PSO2

BCSE0656: IoT FOR INDUSTRIES

Objective: *This course introduce how IoT has become a game changer in the new economy where the Customers are looking for integrated values.*

Credits: 02 **Semester VI** **L–T–P-J: 2–0–0-0**

Module No.	Content	Teaching Hours
I	**Introduction:** Introduction to IoT, What is IIoT? IoT Vs. IIoT, History of IIoT, Basics of industrial IoT, Industrial Sensing and actuation, Cyber Physical System and Next generation Sensors. **IoT Physical Device: Raspberry Pi-** Understanding Pi hardware, Installation of software tools, Connecting with Pi hardware and Running existing programs **M2M protocol:** Open Mobile Alliance Lightweight Machine to Machine Protocol(OMALWM2M)	14
II	**Smart Automotive, Logistics and Supply Chain Via IoT:** Introduction to Automotive Domain, Infotainment Systems- In-Vehicle Infotainment system (IVI), Wire Replacement, Predictive Maintenance, Connected Vehicle Infrastructure-V2V, The Retail Domain, Intelligent Shopping, Energy Management- Smart Building Management, Controlling the Retail Supply Chain, Controlling payments using IoT, Analytics for increasing Store Insights, IoT in Logistics, Tracking Quality of the Shipment, Storage incompatibility detection, Fleet Tracking, **Industrial IoT- Application Domains:** Healthcare, Power Plants, Oil, chemical and pharmaceutical industry, Agriculture and Sports **Real case studies:** Smart City implementation using IoT	14

Text Books:

Reference Books:

- IoT for Industries, Edition 1.0, May 2020, IBM Corporation.
- Michahelles, "Architecting the Internet of Things", ISBN 978-3- 642-19156-5 e-ISBN 978-3-642-19157-2, Springer

Outcome: After completion of the course, the student will be able to:

- CO1: Describe IoT and IIoT
- CO2: Understand the main characteristics of next generation industrial sensors.
- CO3: Understand, design and develop the real life IIoT applications.

Mapping of Course Outcomes (COs) with Program Outcomes (POs) and Program Specific Outcomes (PSOs):

COs	POs/PSOs
CO1	PO10/PSO2
CO2	PO10/PSO1
CO3	PO1,PO10/PSO2
CO4	PO1,PO10/PSO1
CO5	PO5/PSO2
CO6	PO10/PSO2