



**JECRC**<sup>TM</sup>  
**UNIVERSITY**  
BUILD YOUR WORLD

**Faculty of IT & Computer Application**  
**Scheme & Syllabi**  
**of**  
**Bachelor of Computer Applications (BCA)**  
**(Specialization in Data Science and Data Analytics)**  
**(In association with Samatrix)**

**Academic Programme**

**July 2020-21**

**Total Credits for the Batch 2020-23 = 168 Credits**

- 1. Minimum Credits required = 155 Credits**
- 2. Total Relaxation = 13 Credits**
- 3. No relaxation in Core and Foundation subjects**
- 4. Theory exams duration will be of 03 hours and Practical exams will be of 02 hours.**
- 5. Internal Assessment will be of 50 marks and End Term Assessment will be of 50 marks for both lab and theory courses.**

<b>Semester wise credit</b>						<b>Credits</b>	<b>Minimum Credit for degree</b>
<b>I</b>	<b>II</b>	<b>III</b>	<b>IV</b>	<b>V</b>	<b>VI</b>	<b>Total</b>	
<b>28</b>	<b>28</b>	<b>28</b>	<b>28</b>	<b>28</b>	<b>28</b>	<b>168</b>	<b>155</b>

## Semester – I

Course Code	Course Name	L (Hr.)	T (Hr.)	Pr (Hr.)	Credits	Type
BCA101A	Fundamentals of Computer Systems	3	0	0	3	F
BCA102A	Programming Fundamentals using C-I	3	1	0	4	CORE
BCA104A	Digital Electronics	3	0	0	3	F
BCA105B	Communication Skills	3	0	0	3	F
BCA106A	Office Automation Tools Lab	0	0	2	2	F
BCA107A	Programming Fundamentals using C-I Lab	0	0	2	2	CORE
BCA108B	Communication Skills Lab	0	0	2	2	F
BCA166A	Designing Lab( Photoshop and CorelDraw)	0	0	2	2	S
BCA110A	Digital Electronics Lab	0	0	2	2	F
BCA111A	Seminar	0	0	2	2	S
<b>BCADS101 (Data Science Specialization)</b>	Overview of Data Science and Foundation of Data Analysis	3	0	0	3	CORE
	<b>Total</b>	<b>15</b>	<b>1</b>	<b>12</b>	<b>28</b>	

❖ **F- Foundation, ID- Interdisciplinary, S- Specialization (Skill Enhance Courses)**

❖ **L- Lecture, T- Tutorial, P- Practical**

## Semester – II

Course Code	Course Name	L (Hr.)	T (Hr.)	Pr (Hr.)	Credits	Type
BCA112A	Environmental Studies	2	0	0	2	F
BCA113B	Operating System	3	1	0	4	CORE
BCA114B	Database Management System	3	0	0	3	CORE
BCA115B	Computer Architecture	3	1	0	4	F
BCA116A	Programming Fundamentals using C-II	3	1	0	4	CORE
BCA118B	Database Management System Lab	0	0	2	2	CORE
BCA119A	Programming Fundamentals using C-II Lab	0	0	2	2	CORE
BCA120B	HTML Lab	0	0	2	2	F
BCA121A	UNIX and Shell Programming Lab	0	0	2	2	CORE
<b>BCADS102</b> <b>(Data Science</b> <b>Specialization)</b>	Data Analysis Using Python	3	0	0	3	CORE
	<b>Total</b>	<b>17</b>	<b>3</b>	<b>8</b>	<b>28</b>	

❖ **F- Foundation, ID- Interdisciplinary, S- Specialization (Skill Enhance Courses)**

❖ **L- Lecture, T- Tutorial, P- Practical**

## Semester – III

Course Code	Course Name	L (Hr.)	T (Hr.)	Pr (Hr.)	Credits	Type
BCA135B	Computer Network	3	0	0	3	CORE
BCA123A	Programming in PHP	3	0	0	3	S
BCA125A	Data Structures and Algorithms using C	3	1	0	4	CORE
BCA126A	Programming Fundamentals using C++	3	1	0	4	CORE
BCA128A	PHP Lab	0	0	2	2	S
BCA129A	Data Structure and Algorithm using C Lab	0	0	2	2	CORE
BCA130A	Programming Fundamentals using C++ Lab	0	0	2	2	CORE
BCA167A	Multimedia Lab ( Audio/Video editing tools, Flash, Dreamweaver)	0	0	2	2	S
<b>BCADS103</b> <b>(Data Science</b> <b>Specialization)</b>	Probabilistic Modeling and Reasoning	3	0	0	3	CORE
<b>BCADS104</b> <b>(Data Science</b> <b>Specialization)</b>	R Programming for Data Science and Analysis	3	0	0	3	CORE
	<b>Total</b>	<b>18</b>	<b>2</b>	<b>8</b>	<b>28</b>	

❖ **F- Foundation, ID- Interdisciplinary, S- Specialization (Skill Enhance Courses)**

❖ **L- Lecture, T- Tutorial, P- Practical**

## Semester – IV

Course Code	Course Name	L (Hr.)	T (Hr.)	Pr (Hr.)	Credits	Type
BCA171A	Soft Skills	2	0	0	2	F
BCA133A	Programming in Java	3	1	0	4	CORE
BCA134A	Programming in Python	2	1	0	3	S
BCA169A	Data Science & Analytics	3	0	0	3	S
BCA136A	Programming in Java Lab	0	0	2	2	CORE
BCA137A	Programming in Python Lab	0	0	2	2	S
BCA170A	Data Science & Analytics Lab	0	0	2	2	S
<b>BCADS105 (Data Science Specialization)</b>	Foundation of Machine Learning and Pattern Recognition	3	0	0	3	CORE
<b>BCADS106 (Data Science Specialization)</b>	Data Visualization using Tableau or other advance-Tools & Techniques	3	0	0	3	CORE
<b>BCADS109 (Data Science Specialization)</b>	Project	0	0	4	4	S
	<b>Total</b>	<b>16</b>	<b>2</b>	<b>10</b>	<b>28</b>	

❖ F- Foundation, ID- Interdisciplinary, S- Specialization (Skill Enhance Courses)

❖ L- Lecture, T- Tutorial, P- Practical

## Semester – V

Course Code	Course Name	L (Hr.)	T (Hr.)	P (Hr.)	Credits	Type
BCA132A	Introduction to Accounting and Economics	3	0	0	3	ID
BCA141A	Android Applications Development	3	0	0	3	S
BCA142A	Information Security and Cyber Law	3	1	0	4	CORE
BCA143A	Advance Java	3	1	0	4	CORE
BCA145A	Android Applications Development Lab	0	0	2	2	S
BCA146A	Advance Java Lab	0	0	2	2	CORE
BCA138A	Introduction to PL/SQL Lab	0	0	2	2	S
BCA147A	Project	0	0	2	2	S
<b>BCADS107 (Data Science Specialization)</b>	Big Data Analytics with Hadoop	3	0	0	3	CORE
<b>BCADS108 (Data Science Specialization)</b>	Scala for Data Science	3	0	0	3	CORE
	<b>Total</b>	<b>18</b>	<b>2</b>	<b>8</b>	<b>28</b>	

❖ **F- Foundation, ID- Interdisciplinary, S- Specialization (Skill Enhance Courses)**

❖ **L- Lecture, T- Tutorial, P- Practical**

## Semester – VI

Course Code	Course Name	Credits	Type
BCA165A	Industrial Training/Internship/Project Presentation	28	CORE

# Semester I

**Course Name: Fundamentals of Computer Systems**

**Course Code: BCA101A**

<b>L (Hr.)</b>	<b>T (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	0	0	3

## **Course Objectives:**

1. To be able to learn the major components of a computer system
2. To know the Student to learn different types of input and output devices, memories in computer.
3. To enhance the practical knowledge of the students related to Operating system, Computer Software and Boolean algebra.
4. To demonstrate the concept of Logic gates, Simplification of Boolean Expressions Using K-Map and Internet.
5. To know the Characteristics, Generations of computers, Classification of Computers and basic functionality of computers.

## **Syllabus**

### **Unit I**

**Introduction to Computers & Number Systems:** Introduction of Computers, Characteristics, Generations of computers, Classification of Computers, Basic Computer Organization, Applications of Computers, Binary Number System, Working with Binary Numbers, Octal Number System, Hexadecimal Number System, BCD Code ,Other Codes.

### **Unit II**

**Input-Output devices & Computer Memory:** Input Devices, Output Devices, Soft Copy Devices, Hard Copy Devices, Memory Type: Memory Hierarchy, Processor Registers, Cache Memory, Primary Memory, Secondary Storage Devices, Magnetic Tapes, Floppy Disks, Hard Disks, Optical Drives, USB Flash Drives , Memory Cards, Mass Storage Devices



### **Unit III**

**Computer Software & Operating Systems:** Computer Software: Introduction to Computer Software, Classification , System Software, Application Software , Acquiring Computer Software, Productivity Software, Graphics Software, Database management software .

Operating Systems: Basic Definitions, History, Characteristics, Properties, Classifications, Functions of OS, Popular Operating Systems, mobile operating System.

### **Unit IV**

**Boolean Algebra and Logic Gates:** Boolean Algebra ,Venn Diagrams ,Truth Tables ,Basic Laws of Boolean Algebra, Representations of Boolean Functions, Logic Gates, Logic Diagrams and Boolean Expressions ,Universal Gates, Simplification of Boolean Expressions Using K-Map.

### **Unit V**

**Computer Networks and Internet:** Computer Networks: Introduction to Computer Networks, Connecting Media, Network Topologies, Types of Network, Networking Devices, Wireless Networks, Data Transmission Mode, Open System Interconnection Model, TCP/IP Model, Internet, Internet Protocol Address., DNS, URL.

The Internet & Emerging Computer Technologies: History, Internet Services, Internet Glossary, Types of Internet Connections, Internet Security, Distributed Networking, Peer-to-peer Computing, Grid Computing, Cloud Computing, Utility Computing, Wireless Network, Bluetooth, Artificial Intelligence

### **Course Outcomes (COs):**

**Upon successful completion of this subject students should be able to:**

CO1: Understand what is computer system and Basic information related to computer.

CO2: Understand the fundamental hardware components that make up a computer's hardware and the role of each of these components

CO3: Understand the difference between an operating system and an application program, and what each is used for in a computer

CO4: Understand the fundamentals of Logic gates and circuit design

CO5: Understand basics of World Wide Web and new Internet technologies.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

Course Outcome	Program Outcome											
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1				H						H		H
CO2			H			H	M					
CO3		H										
CO4							H	H				
CO5	H				M						L	

H = Highly Related; M = Medium; L = Low

**Text Books**

1. Madan, S. , “Fundamentals of Computers and Information System”, Scholar Tech Press, 2012
2. Reema Theraja , "Fundamentals of Computers", Oxford University Press, 2014

**Reference Books**

1. Mohan, P., “Fundamentals of Computers”, Himalaya publishing House, 2010
2. Murthy, C. S., “Fundamentals of Computers”, Himalaya Publishing House, 2010
3. Tiwari, H. N., & Jain, H. C., “Fundamentals of Computer and Information System”, International Book House Pvt. Ltd., 2012

**Course Name: Programming Fundamentals using C-I**

**Course Code: BCA102A**

<b>L (Hr.)</b>	<b>T (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	1	0	4

**Course Objectives:**

1. To provide exposure to problem solving through programming.
2. To enhance the ability to solve the problem through various constructs of C such as decision control and looping statements.
3. To demonstrate the concept of Accessing Array Elements, Storing Values in Arrays and length of an Array.
4. To evaluate the concept of data types and variable.
5. To explore major concepts of computer science and the process of computer programming, including programming, procedural and data abstraction and program modularity.

**Syllabus**

**Unit I**

**Introduction to Programming and C:** Introduction to Computer Software, Classification of Computer Software, Programming Languages, Generation of Programming Languages  
C Programming Language: Introduction, Structure of C Program, Files Used in C Program,

**Unit II**

**Data types and Variables:** Compiling and Executing C Programs, Keywords, Identifiers , Data Types, Variables, Constants, Input/output Statement in C, Operators in C, Type Conversion and Typecasting, Structure of C Program, Operators: Arithmetic operators, relational and logical operators.

**Unit III**

**Decision Control and Looping Statements:** Decision Control Statements, Conditional Branching Statements, Iterative Statements, Nested Loops, break and continue Statements, goto Statement, Building a Robust C Program, Flow Chart Designing

#### Unit IV

**Array:** Declaration of Arrays, Accessing Array Elements, Storing Values in Arrays, Length of an Array, Array Operations, Multidimensional Arrays, Two-Dimensional Arrays, and Operations on Two-dimensional Arrays.

#### Unit V

**String:** Introduction, Suppressing Input, String Operations, String and Character Functions, String Array.

#### Course Outcomes (COs):

**Upon successful completion of this subject students should be able to:**

CO1: Students will learn the concept of character set identifiers and keywords, data type, variable names, declaration statements and basic structure of C program.

CO2: Student will classify the String Operations, String and Character Functions.

CO3: Students will apply the concept of Arithmetic operators, relational and logical operators, type, conversion, Standard input and output, formatted output and input.

CO4: Students will contrast C programs based on Flow of Control and program Structures.

CO5: Students will learn the concept of Accessing Array Elements, Storing Values in Arrays, and length of an Array

#### MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome											
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1				H						H		H
CO2			H			H	M					
CO3		H										
CO4							H	H				
CO5	H				M						L	

H = Highly Related; M = Medium; L = Low

**Text Books**

1. Reema Theraja , "Programming in C", Second Edition , Oxford University Press, 2016
2. Yeshwant kanetkar, Let us C, Fifteenth Edition, BPB Publication 2016.

**Reference Books**

1. Reema Thareja, "Introduction to C Programming", First Edition, Oxford University press, 2012.
2. Byron S Gottfriend, "Programming with C", Second Edition, Mc Graw Hill, 2007.
3. Herbert Schildt, Turbo C: The Complete Reference, Fourth edition, Mc Graw-Hill, 2017.

**Course Name: Digital Electronics**

**Course Code: BCA104A**

<b>L (Hr.)</b>	<b>T (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	0	0	3

**Course Objective:**

1. To introduce the concept a comprehensive introduction to digital logic design leading to the ability to understand number system representations.
2. To know binary codes, binary arithmetic and Boolean algebra, its axioms and theorems, and its relevance to digital logic design.
3. To know about synchronous sequential logic and Asynchronous sequential logic.
4. To acquire the basic knowledge of digital logic levels and application of knowledge to understand digital electronics circuits.
5. To prepare students to perform the analysis and design of various digital electronic circuits.

**Syllabus**

**Unit I**

**Binary Systems:** Digital Systems, Binary Numbers, Number Base Conversions, Octal and Hexadecimal Numbers, Complements, Signed Binary Numbers, Binary Codes, Binary Storage and Registers, Binary Logic.

**Unit II**

**Boolean Algebra and Logic Gates:** Basic Definitions, Axiomatic Definition of Boolean Algebra, Basic Theorems and Properties of Boolean Algebra, Boolean Functions, Canonical and Standard Forms, Other Logic Operating, Digital Logic Operations, Digital Logic Gates, Integrated Circuits.

**Unit III**

**Simplification of Boolean Functions:** Karnaugh map and Quine-Mccluskey methods, SOP, POS simplification, NAND and NOR implementations Don't - Care Conditions, Exclusive - OR Function.



CO1			H				M		M			
CO2		H						M				
CO3					H	M						
CO4				H				M				
CO5	M								M		L	

H = Highly Related; M = Medium; L = Low

### Text Books

1. M Morris Mano, “ Digital Design” , 3rd Edition, 2006, PHI
2. Bignell& Donovan Digital Electronics, 4th Edition, 2007, Thomson Learning.

### Reference Books

1. R. P Jain, Modern Digital Electronics, Second Edition, TMH
2. L K Maheswari and M M S Anand, “Laboratory Manual for Introductory Electronic Experiments”, New Age, 2010.
3. S Poornachandra Rao and B Sasikala, “Handbook of Experiments in Electronics and Communication Engineering”,Vikas publishers, 2003.



**Course Name: Communication Skills**

**Course code: BCA105A**

<b>L (Hr.)</b>	<b>T (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	0	0	3

**Course Objectives:**

1. To understand the basic concepts of three major forms of communications which are vital in academic and professional settings namely professional presentations, interviews and group communications respectively.
2. To have a deep insight into the techniques for delivering effective presentations.
3. To have a deep insight for winning job interviews and actively participating in various forms of group communication.
4. To apply the critical and creative thinking abilities necessary for effective communication in today's business world.
5. To summarize the better writing skills by briefing the learners to the dynamics of effective writing.

**Syllabus**

**Unit I**

**Essentials of Grammar**

1. Parts of Speech
2. Articles
3. Question Tags
4. Conditional Sentences
5. Modal Verbs

**Unit II**

**Applied Grammar**

1. Tenses
2. Passive Voice
3. Indirect Speech
4. Relative Clauses

### **Unit III**

#### **Composition**

Dialogue Writing

Paragraph Writing

Precise Writing

Report, its importance and Report Writing

### **Unit IV**

#### **Poems**

1. The Character of A Happy Life: Sir Henry Wotton
2. No Men are Foreign: James Kirkup
3. If : Rudyard Kipling

### **Unit V**

#### **Short Stories**

1. How Much Land Does a Man Need?: Leo Tolstoy
2. The Last Leaf: O. Henry

#### **Course Outcomes (COs):**

##### **Upon successful completion of this subject students should be able to:**

CO1: Understand the communication processes and practices in academic and professional contexts.

CO2: Able to understand the concept and basics of communication and essentials of grammar.

CO3: Students are able to work with jumble words and grammar.

CO4: Enhance the knowledge of composition; produce written academic and professional documents.

CO5: Have the capability to apply an ability to properly carry out the purpose and significance of the channels of Communication and improve writing skills for stories and poems.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

Course Outcome	Program Outcome											
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	M	H									M	
CO2					M							H
CO3								H	M			
CO4									H		M	
CO5	H									H		

H = Highly Related; M = Medium; L = Low

**Text Books**

1. Sanjay Kumar & Pushp Lata , "Communication Skills ", Second Edition, Oxford University Press.
2. English for Engineers: Made Easy, AedaAbidi&RituChaudhary, Cengage Learning, (New Delhi)

**Reference Books**

1. A Practical Course for Developing Writing Skills in English, J.K. Gangal, PHI Learning Pvt. Ltd., New Delhi.
2. Intermediate Grammar, Usage and Composition, Tickoo, A. E. Subramaniam& P. R.Subramaniam, Orient Longman (New Delhi)
3. The Written Word , Vandana R. Singh, Oxford University Press (New Delhi)R.T.U., Kota Scheme and

**Course Name: Office Automation Tools Lab**

**Course Code: BCA106A**

<b>L (Hr.)</b>	<b>T (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
0	0	2	2

**Course Objectives:**

The purpose of this course is to enhance the practical knowledge based on prescribed theory course. The students will be able to enhance their analyzing and problem solving skills after implementation of all the given experiments.

**List of Sample Programs**

Elaborate these topics in with the help of following tools ie. open office/MS office/Libre office

**1. Introduction**

- 1.1. Concept of Windows, Icon, Menu
- 1.2. Desktop
- 1.3. Creating Folders and Shortcuts
- 1.4. Finding Files & Folders
- 1.5. Creating, Copying, Moving and Deleting files
- 1.6. Windows Explorer

**2. Word Processing Package**

- 2.1. Typing, Editing, Proofing & reviewing
- 2.2. Formatting text & Paragraph
- 2.3. Automatics Formatting and Styles
- 2.4. Working with Tables
- 2.5. Graphics and Sound
- 2.6. Mail Merge

**3. Spreadsheet package**

- 3.1. Concept of worksheet
- 3.2. Working & Editing in Workbooks
- 3.3. Creating Formats & Links

- 3.4. Protecting and Hiding data
- 3.5. Built in Functions (Mathematical, Statistical, String & Date)
- 3.6. Formatting a Worksheet & Creating graphics objects
- 3.7. Creating Charts (Graphics), Formatting and analyzing data
- 3.8. Sharing & Importing Data
- 3.9. Printing

#### **4. Presentation Package**

- 4.1. Creating and Editing Slides
- 4.2. Creating and Editing objects in the slide
- 4.3. Animation
- 4.4. Creating and Running Slide Show
- 4.5. Templates

#### **5. Internet**

- 5.1. Concepts
- 5.2. Working
- 5.3. Mailing & surfing tools

**Course Name: Programming Fundamentals using C-I Lab**

**Course Code: BCA107A**

<b>L (Hr.)</b>	<b>T (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
0	0	2	2

**Course Objectives:**

The purpose of this course is to enhance the practical knowledge based on prescribed theory course. The students will be able to enhance their analyzing and problem solving skills after implementation of all the given experiments.

**List of Sample Programs**

1. Write a program to demonstrate the use of printf Statement to print values of variables of different data types.
2. Write a program to calculate the area of a triangle using Hero's formula.
3. Write a program to calculate the distance between two points.
4. Write a program to perform addition, Subtraction Division, Integer Division, Multiplication and modulo Division on two Integer numbers.
5. Write a program to subtract two long Integers.
6. Write a program to illustrate the use of Unary prefix increment and decrement operators.
7. Write a program to find the largest of three numbers using Ternary operators.
8. Write a program to show use of Bitwise operators.
9. Write a program to demonstrate the use of Assignment Operators
10. Write a program to Swap 2 numbers using a temporary variable.
11. Write a program to Swap 2 numbers without using a temporary variable.
12. Write a program to convert degrees fahrenheit into Degrees Celsius.
13. Write a program to convert a floating point number into the corresponding integer.
14. Write a program to convert an Integer into the Corresponding floating point number.
15. Write a program to determine whether a person is eligible to vote or not.
16. Write a program to determine the character entered by the user.
17. Write a program to find whether the given number is even or odd.
18. Write a program to find whether the given year is a leap year or not.
19. Write a program to test whether a number entered is positive , negative or equal to zero.

20. Write a program to input three numbers and then find largest of them using && operator.
21. Write a program to calculate the roots of a quadratic equation.
22. Write a program to demonstrate the use of switch statement without a break.
23. Write a program that accepts a number from 1 to 10. Print whether the number is even or odd using a switch case construct.
24. Write a program to print the following pattern
 

```

      *****
      *****
      *****
      *****
      
```
25. Write a program to print the following pattern
 

```

      A
      AB
      ABC
      ABCD
      ABCDE
      
```
26. Write a program to print the Fibonacci series using recursion.
27. Write a program to read and display n numbers using an array.
28. Write a program to print the position of the smallest of a number using arrays.
29. Write a program to enter n number of digits. Form a number using these digits.
30. Write a program to transpose a 3X3 matrix.
31. Write a program to read and display a 3X3 matrix.
32. Write a program to print the elements of a 2D array.
33. Write a program to display a string using printf ().
34. Write a program to find the length of a string.
35. Write a program to convert characters of a string into lower case.

**Course Name: Communication Skills Lab**

**Course Code: BCA108A**

L (Hr.)	T (Hr.)	Pr (Hr.)	Credits
---------	---------	----------	---------

0	0	2	2
---	---	---	---

### **Course Objectives:**

The purpose of this course is to enhance the practical knowledge based on prescribed theory course. The students will be able to enhance their analyzing and problem solving skills after implementation of all the given experiments.

### **List of Sample Programs**

#### **Building Advanced Vocabulary**

1. Word Formation
2. Affixes
3. Words often Mis-spelt and Mis- Pronounced
4. Words often Confused
5. Homonyms and Homophones
6. One Word for Many.
7. Synonyms and Antonyms.
8. Jumbled Sentences
9. Practice Exercise of Linking Words (Conjunctions) on software
10. Practice Exercise of Tenses on soft ware
11. Practice Exercise of passive voice on software
12. Dialogue conversation Practice on software

**Course Name: Designing Lab ( Photoshop and CorelDraw)**

**Course Code: BCA166A**

<b>L (Hr.)</b>	<b>T (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
0	0	2	2



**Course Objectives:**

The purpose of this course is to enhance the practical knowledge based on prescribed theory course. The students will be able to enhance their analyzing and problem solving skills after implementation of all the given experiments.

**List of Sample Programs**

1. This exercise can be solved using the following functions: Selection Tools, Copy, Cut, Paste, Move Tool.
2. This exercise can be solved using the following functions: Lasso- and Polygonal Lasso Selection Tools, Copy, Paste Into, Move Tool, Zoom Tool, Transform.
3. This exercise can be solved using the following functions: Quick Select Tool (or Magic Wand Tool), Invert Selection, Copy, Paste.
4. This exercise can be solved using the following functions: Paint Bucket Tool, Color Picker, Brush Tool.
5. This exercise can be solved using the following functions: New Picture, Brush Tool, Layers, Eraser Tool.
6. Creating the master page in CorelDraw
7. Creating pie charts in CorelDraw
8. Creating flow charts in CorelDraw
9. Creating interactive navigational buttons in CorelDraw
10. Publishing your presentation as a full screen PDF in CorelDraw

**Course Name: Digital Electronics Lab**

**Course Code: BCA110A**

<b>L (Hr.)</b>	<b>T (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
0	0	2	2

**Course Objectives:**

The purpose of this course is to enhance the practical knowledge based on prescribed theory course. The students will be able to enhance their analyzing and problem solving skills after implementation of all the given experiments.

### List of Sample Programs

1. To study and verify the truth table of logic gates.
2. Design and implementation of Adder and Subtractor using logic gates.
3. Design and implementation of BCD to excess-3 code converter using logic gates.
4. Design and implementation of Binary to gray code converter using logic gates.
5. Design and implementation of 4 bit binary Adder/ subtractor using IC 7483
6. Design and implementation of 4 bit binary BCD adder using IC 7483
7. Design and implementation of 2 bit Magnitude Comparator using logic gates.
8. Design and implementation of 16 bit odd/even parity checker generator using IC74180.
9. Design and implementation of multiplexer using logic gates, IC74150 and IC74154.
10. Design and implementation of De-multiplexer using logic gates, IC74150 and IC74154
11. Design and implementation of encoder using logic gates, IC7445 and IC74147
12. Design and implementation of decoder using logic gates, IC7445 and IC74147
13. Construction and verification of 4 bit ripple counter.
14. Design and implementation of 3-bit synchronous up/down counter.
15. Implementation of SISO, SIPO, PISO and PIPO shift registers using Flip- flops

**Course Name: Seminar**

**Course Code: BCA111A**

<b>L (Hr.)</b>	<b>T (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
0	0	2	2

**Course Objective:**

The students will present a seminar on a technical topic in front of class and faculty. The evaluation will be done on individual performance followed by submission of a written report on Seminar Topic.

## **SEMESTER II**

**Course Name: Environmental Studies**

**Course Code: BCA112A**

<b>L (Hr.)</b>	<b>T (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
2	0	0	2

### **Course Objectives**

1. To define and use correctly the common terms of environmental science.

2. To explain what makes up the environment, how it functions, and how humans are part of it.
3. To apply the concepts and principles of environmental science to propose solutions to specific environmental problems.
4. To analysis environmental writings and predictions and their impact on subsequent developments in human relationship with the environment.
5. To evaluate the adequacy of conclusions about environmental phenomena.

## **Syllabus**

### **Unit I**

**Introduction and Natural Resources:** Multidisciplinary nature and public awareness, Renewable and nonrenewal resources and associated problems, Forest resources, Water resources, Mineral resources, Food resources, Energy resources, Land resources, Conservation of natural resources and human role.

### **Unit II**

**Ecosystems:** Concept, Structure and function, Producers composers and decomposers, Energy flow, Ecological succession, Food chains webs and ecological pyramids, Characteristics structures and functions of ecosystems such as Forest, Grassland, Desert, Aquatic ecosystems.

### **Unit III**

**Biodiversity and Conservation:** Definition, Genetic, Species, and Ecosystem diversity, Bio-geographical classification of India, Value of biodiversity at global, national, local levels, India as a mega diversity nation, Hot sports of biodiversity, Threats to biodiversity, Endangered and endemic species of India, In-situ and ex-situ conservation of biodiversity.

### **Unit IV**

**Environmental Pollution:** Definition, Causes, effects and control of air pollution, water pollution, soil pollution, marine pollution, noise pollution, thermal pollution, nuclear hazards, human role in prevention of pollution, Solid waste management, Disaster management, floods, earthquake, cyclone and landslides.

### **Unit V**

**Social issues and Environment:** Unsustainable to sustainable development, Urban problems related to energy, Water conservation and watershed management, Resettlement and re-habitation, Ethics, Climate change, Global warming, Acid rain, Ozone layer depletion, Nuclear accidents, holocaust, Waste land reclamation, Consumerism and waste products, Environment protection act, Wildlife protection act, Forest conservation act, Environmental issues in legislation, population explosion and family welfare program, Environment and human health, HIV, Women and child welfare, Role of information technology in environment and human health.

**Course Outcomes:**

**Upon successful completion of this subject students should be able to:**

CO1: Understand fundamental physical and biological principles that govern natural processes.

CO2: Understand fundamental concepts from the social sciences and the humanities underlying environmental thought and governance.

CO3: Integrate and apply perspectives from across the natural sciences, social sciences, and the humanities in the context of complex environmental problems.

CO4: Apply the concepts and principles of environmental science to propose solutions to specific environmental problems.

CO5: Able to understand social and environmental issues.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

Course Outcome	Program Outcome											
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1				H						H		H
CO2			H			H	M					
CO3		H										
CO4							H	H				
CO5	H				M						L	

H = Highly Related; M = Medium; L = Low

### **Text Books**

1. Rajgopalan , “Environmental Studies From Crisis to Cure” , (3e), Oxford University Press.
2. Agarwal, K.C., Environmental Biology, Nidi Publication Ltd., Bikaner, 2001.

### **Reference Books**

1. BharuchaErach, Biodiversity of India, Mapin Publishing Pvt. Ltd., Ahmadabad, 2002.
2. Clark, R.S., Marine Pollution, Clanderson Press, Oxford, 2002.
3. Cunningham, W.P., et al., Environmental Encyclopedia, Jaico Publishing House, Mumbai, 2003.

**Course Name: Operating System**

**Course Code: BCA113B**

<b>L (Hr.)</b>	<b>T (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	1	0	4

### **Course Objectives:**

1. To understand the OS role in the overall computer system
2. To study the operations performed by OS as a resource manager and the scheduling policies of OS
3. To understand the different memory management techniques
4. To understand process concurrency and synchronization
5. To understand the concepts of input/output, storage and file management and to study different OS and compare their features.

### **Syllabus**

#### **Unit – I**

**Introduction** - Operating system objectives, User view, System view, Operating system definition ,Types, Functions, Computer System Organization, Computer System Architecture, OS Structure,

Operating System services, User and OS Interface, System Programs, Operating System Design and Implementation, OS Structure.

## **Unit – II**

**Process and CPU Scheduling** – Process concepts, The Process, Process State, Process Control Block, Threads, Process Scheduling, Scheduling Criteria, Scheduling algorithms Scheduling Queues, Schedulers, Context Switch, Operations on Processes, System calls, Process Synchronization, The Critical Section Problem, Peterson’s solution, Synchronization Hardware, Semaphores.

## **Unit – III**

**Deadlocks-** Deadlock characterization, Methods of handling deadlock, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery from Deadlock.

## **Unit - IV**

**Memory Management** – Memory Management Strategies- Overlays and Swapping, Contiguous Memory Allocation, Segmentation, Paging, Structure of Page Table, Virtual Memory Management- Segmentation, Paging, Demand Paging, Page Replacement, Page Replacement Algorithms, Allocation of Frames, Thrashing.

## **Unit - V**

**File System and Security** - File-System Structure, File-System Implementation , Directory Implementation, Allocation Methods, Free-Space Management, Efficiency and Performance, Recovery, NFS, Organization of I/O Function, I/O Buffering, Disk Organization, Disk Scheduling, RAID, Operating Systems view of file system, Disk space management. System Security- The Security Problem, Program Threats, Policies, System and Network Threats, User Authentication, Security Models

## **Course Outcomes (COs):**

**Upon successful completion of this subject students should be able to:**

CO1: Apply optimization techniques for the improvement of system performance.

CO2: Ability to design and solve synchronization problems.

CO3 Learn about minimization of turnaround time, waiting time and response time and also maximization of throughput by keeping CPU as busy as possible.

CO4: Ability to change access controls to protect files.

CO5: Ability to compare the different operating systems.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

Course Outcome	Program Outcome											
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1		H							M			
CO2				H								
CO3			H				M				M	
CO4			M									H
CO5		M			H					H		

H = Highly Related; M = Medium; L = Low

**Text Books**

1. A Silberschatz, P.B. Galvin, G. Gagne, Operating Systems Concepts, 8<sup>th</sup> edition, John Wiley Publications ,2008.
2. A.S. Tanenbaum, Modern Operating Systems, 3<sup>rd</sup> edition, Pearson Education ,2007.

**Reference Books**

1. G. Nutt, Operating Systems: A Modern Perspective, 2<sup>nd</sup> edition Pearson Education ,1997.
2. W. Stallings, Operating Systems, Internals & Design Principles, 5<sup>th</sup> edition, Prentice Hall of India, 2008.
3. “Operating Systems: Internals and Design Principles” by William Stallings



**Course Name: Database Management System**

**Course Code: BCA114B**

<b>L (Hr.)</b>	<b>T (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	1	0	4

**Course Objectives:**

1. To enhance the fundamentals knowledge of data models and to conceptualize and depict a database system using ER diagram.
2. To develop programs and queries based on SQL and relational database, and do fundamental Operations of Relational Algebra & their Implementation.
3. To be able to know the fundamental concepts of transaction processing management.
4. To justify the concept of concurrency control techniques and recovery procedure.
5. To have an introductory knowledge about the Storage and Query processing Techniques.

**Syllabus**

**Unit I**

**Basic Concepts:** Purpose of database systems-Components of DBMS – DBMS Architecture and Data Independence- Data modeling, Entity Relationship Model, Relational, Network, Hierarchical and object oriented models, Data Modeling using the Entity Relationship Model.

**Unit II**

**Structure of relational databases:** Relational Constraints, Domain Constraints, Key Constraints Referential Integrity Constraints, Relational Algebra, Fundamental Operations of Relational Algebra & their Implementation, Data definition with SQL, insert, delete and update statements in SQL – views – data manipulation with SQL.

### **Unit III**

**Query Processing:** Methods for Joining Tables –Nested Loop Join Merge Join, Hybrid Join, Multiple table Join, Transforming Nested Queries to Joins, Object Relational SQL, Procedural SQL, Introduction to Embedded SQL.

### **Unit IV**

**Database Design:** Integrity Constraints – Domain Constraints- Referential integrity – Functional Dependency- Normalization using Functional Dependencies, Normal forms based on primary keys- general definitions of Second and Third Normal Forms. Boyce Codd Normal Form– Multivalued Dependencies and Forth Normal Form – Join Dependencies and Fifth Normal Form.

### **Unit V**

**Transaction Management:** Transaction Concept, ACID Properties, Transaction State, Implementation of ACID properties, Schedules. Concurrency Control: Need of concurrency control, Concurrency control techniques, Lock based protocols, binary lock, share and exclusive lock, two phase locking protocol. Introduction to recovery.

### **Course Outcomes:**

**Upon successful completion of this subject students should be able to:**

CO1: Recall the basic concept of DBMS and models of DBMS.

CO2: To know the fundamental concepts of transaction processing management.

CO3: To develop programs and queries based on SQL and relational database, and do fundamental Operations of Relational Algebra & their Implementation.

CO4: Contrast the concept of functional dependency, Norm forms, constraints and integrity

CO5: Able to understand the concept of concurrency control techniques and recovery.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

Course Outcome	Program Outcome											
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H		M						M			M
CO2		H			H							
CO3			M	H				M				
CO4						M					M	H
CO5	M						H					

H = Highly Related; M = Medium; L = Low

**Text Books**

1. Fundamentals of Database System Elmasri and Navathe (4rd Edition), Pearson Education Asia (2008)
2. An Introduction to Database Systems - C.J.Date (7th Edition) Pearson Education Asia (2006)

**Reference Books**

1. A.Silberschatz, H. Korth and S. Sudarshan, *Database System Concepts*, 5th Edition, McGraw Hill, 2010.
2. R. Ramakrishnan, J. Gehrke, *Database Management Systems*, 3<sup>rd</sup> edition, McGraw Hill International Edition, 2007.
3. Database System Concepts - Henry F Korth, Abraham Silbershatz, McGraw Hill 2nd edition. (2005)

**Course Name: Computer Architecture**

**Course Code: BCA115B**

L (Hr.)	T (Hr.)	Pr (Hr.)	Credits
3	1	0	4

### Course Objectives:-

1. To provide information about digital computer technology and how data is represented in computer system.
2. To discuss in detail the operation of the arithmetic Unit including the algorithms & implementation of fixed-point and floating-point addition, subtraction, multiplication & division.
3. To have a systematic understanding of the basic structure and operation of a digital computer.
4. To clear concepts of central processing unit and describes the structure of arithmetic/logic units.
5. To understand the pipelining and vector processing concepts.

### Syllabus

#### Unit I

**Digital logic circuits, Digital component and Data Representation:** Digital computers, Logic gates Boolean algebra, Complement of a function, combinational circuits, Half adder, Full Adder, Flip flop: SR flip flop, D Flip flop, JK Flip Flop, T flip flop, integrated circuits, Decoder, Multiplexer, Registers Shift register Binary Counter Memory Unit.

**Data Representation:** Data Types, Number Systems, Octal and Hexadecimal Numbers, Decimal Representation, Alphanumeric Representation, Complements,  $(r-1)$ 's Complement,  $( r's )$  Complement, Subtraction of Unsigned Numbers, Fixed-Point Representation, Integer Representation, Arithmetic Addition, Arithmetic Subtraction.

#### Unit II

#### **Register Transfer and Micro operation, Basic Computer Organization and Design:**

Register Transfer, Bus and Memory Transfer, Arithmetic, Logic and shift Micro operations, Instruction Codes, Stored Program Organization, Indirect Address, Computer Registers, Common Bus System, Computer Instructions, Instruction Set Completeness, Timing and Control, Instruction

Cycle, Fetch and Decode, Determine the Type of Instruction, Register-Reference Instructions, Memory-Reference Instructions.

### **Unit III**

**Central Processing Unit:** General Register Organization, Control Word, Examples of Micro operations, Stack Organization, Register Stack, Memory Stack, Reverse Polish Notation, Evaluation of Arithmetic Expressions, Instruction Formats, Three-Address Instructions, Two-Address Instructions, One-Address Instructions, Zero-Address Instructions, RISC, CISC.

### **Unit IV**

**Pipeline and Vector Processing, Computer Arithmetic:** Parallel Processing, Pipelining, General Considerations, Arithmetic Pipeline, Instruction Pipeline, Four-Segment Instruction Pipeline, Data Dependency, Handling of Branch Instructions, RISC Pipeline, Three-Segment Instruction Pipeline, Delayed Load, Delayed Branch, Vector Processing.

Addition and Subtraction, Addition and Subtraction with Signed-Magnitude Data, Hardware Implementation, Addition and Subtraction with Signed-2's Complement Data, Multiplication Algorithms Hardware Implementation for Signed-Magnitude Data, Booth Multiplication Algorithm, Array Multiplier, Division Algorithms, Floating-Point Arithmetic Operations, Basic Considerations, Register Configuration, Addition and Subtraction, Multiplication, Division, Decimal Arithmetic Unit.

### **Unit V**

#### **DMA and Memory organization**

Direct Memory access, DMA controller, DMA Transfer, Memory Hierarchy, Main Memory, Cache Memory: Associative mapping, Direct mapping, set associative mapping, writing into Cache, Cache initialization, Virtual Memory: Address space and memory space, address mapping using pages, associative memory page table, Page replacement algorithms.

#### **Course Outcomes (COs):-**

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

CO1: Understand the concept of digital logic circuit and digital component.

CO2: Understand the concept of register transfer and micro operations.

CO3: Develop an understanding of concepts of central processing unit and describes the structure of arithmetic/logic units.

CO4: Examine the pipelining and vector processing concepts and implement computer arithmetic.

CO5: Understand the concept of DMA and Memory organization

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

Course Outcome	Program Outcome											
	PO 1	PO 2	PO 3	PO 4	PO 5	PO 6	PO 7	PO 8	PO 9	PO1 0	PO1 1	PO1 2
CO1				H					M			
CO2		M		M						M		
CO3	M				M		L					M
CO4			L					M				
CO5	H				L				M			

H = Highly Related; M = Medium; L = Low

**Text Books:**

1. Morris Mano, "Computer system architecture", 5<sup>th</sup> edition, Pearson Education.
2. Carl Hamacher, Zvonko Vranesic and Safwat Zaky, Naraig Manjikian, "Computer Organization and Embedded Systems", (6e), McGraw Hill Publication, 2012

**Reference Books :**

1. D. A. Patterson and J. L. Hennessy, "Computer Organization and Design - The Hardware/Software Interface", (5e), Morgan Kaufmann, 2011
2. Mohammed Rafiquzzaman and Rajan Chandra, "Modern Computer Architecture", Galgotia Publications Pvt. Ltd. 2008.
3. William Stallings, "Computer Organization and Architecture Designing for Performance", (8e), PHI, 2009.

**Course Name: Programming Fundamentals using C-II**

**Course Code: BCA116A**

<b>L (Hr.)</b>	<b>T (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	1	0	4

**Course Objectives:**

1. To provide exposure to problem solving through programming.
2. To enhance the ability to solve the problem through various constructs of C such as decision control and looping statements.
3. To demonstrate the concept of functions, arrays, strings, pointers, structure and union.
4. To evaluate the concept of file management and pre-processor directives.
5. To explore major concepts of computer science and the process of computer programming, including programming, procedural and data abstraction and program modularity.

## **Syllabus**

### **Unit I**

**Introduction to C:** Data types, variables, Decision Control Statements, Conditional Branching Statements, Iterative Statements, Nested Loops, break and continue Statements, Array, and String.

### **Unit II**

**Functions:** Function Declaration/Function Prototype, Definitions, Function call, Return Statement, Parameter passing, Scope of Variables, Storage classes, Recursive functions

### **Unit III**

**Pointers:** Computer's Memory, Introduction to Pointers, Declaring Pointer Variables, Pointer Expressions and Pointer Arithmetic, Null Pointers, Generic Pointers, Passing Arguments, Pointers and Arrays, Pointers and Strings, Array of Pointers, Function Pointers, Array of Function Pointers, Pointers to Pointers

### **Unit IV**

**Structure & Union:** Structure Declarations, Typedef keyword, initialization, Accessing members, nested structure, Array of Structure, Structure with functions, Self-referential Structures, Union, Structure vs Union

### **Unit V**

**Files and Preprocessor Directives:** Files: Introduction to Files, Using Files in C, Writing to files, Read data, EOF, Error handling, Command line Arguments, Functions for Selecting a Record Randomly, Renaming the File, Temporary File.

Preprocessor Directives: Introduction, Types, #define, #include, #undef, #line, Pragma Directives, Conditional Directives, The Defined Operator, #error Directive, Predefined Macro Names.

### **Course Outcomes (COs):**

**Upon successful completion of this subject students should be able to:**

CO1: Students will learn the concept of character set identifiers and keywords, data type, variable names, declaration statements and basic structure of C program and Flow of Control and program Structures.

CO2: Student will classify the File handling, preprocessor directives and predefined macro names.



CO3: Students will apply the concept of Arithmetic operators, relational and logical operators, type, conversion, Standard input and output, formatted output and input.

CO4: Students will contrast C programs based on Structures Union and Files.

CO5: Students will learn the concept of Arrays, Pointers, Null Pointers, and Generic Pointers

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

Course Outcome	Program Outcome											
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1				H						H		H
CO2			H			H	M					
CO3		H										
CO4							H	H				
CO5	H				M						L	

H = Highly Related; M = Medium; L = Low

**Text Books**

1. Reema Theraja , "Programming in C", Second Edition , Oxford University Press, 2016
2. Reema Thareja, "Introduction to C Programming", First Edition, Oxford University press, 2012.

**Reference Books**

1. Yeshwant kanetkar, Let us C, Fifteenth Edition, BPB Publication 2016.
2. Byron S Gottfried, "Programming with C", Second Edition, Mc Graw Hill, 2007.
3. Herbert Schildt, Turbo C: The Complete Reference, Fourth edition, Mc Graw-Hill, 2017.

**Course Name: Database Management System Lab**  
**Course Code: BCA118B**

<b>L (Hr.)</b>	<b>T (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
0	0	2	2

**Course Objectives:**

The purpose of this course is to enhance the practical knowledge based on prescribed theory course. The students will be able to enhance their analyzing and problem solving skills after implementation of all the given experiments.

**List of Sample Programs**

1. Data Definition Language (DDL) commands in DBMS
2. Data Manipulation Language (DML) and Data
3. Control Language (DCL)
4. Join – Inner Join and Outer Join
5. High level language extensions with cursors
6. High level language extension with Triggers
7. Procedures and Functions
8. Embedded SQL
9. Database design using E-R model and Normalization
10. Design and implementation of payroll processing system
11. Design and implementation of banking system

**Course Name: Programming Fundamentals using C-II Lab**

**Course Code: BCA119A**

<b>L (Hr.)</b>	<b>T (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
0	0	2	2

**Course Objectives:**

The purpose of this course is to enhance the practical knowledge based on prescribed theory course. The students will be able to enhance their analyzing and problem solving skills after implementation of all the given experiments.

**List of Sample Programs**

1. Write a program to demonstrate the use of printf Statement to print values of variables of different data types.
2. Write a program to calculate the area of a triangle using Hero's formula.
3. Write a program to calculate the distance between two points.
4. Write a program to perform addition, Subtraction Division, Integer Division, Multiplication and modulo Division on two Integer numbers.
5. Write a program to subtract two long Integers.
6. Write a program to illustrate the use of Unary prefix increment and decrement operators.
7. Write a program to find the largest of three numbers using Ternary operators.
8. Write a program to show use of Bitwise operators.
9. Write a program to demonstrate the use of Assignment Operators
10. Write a program to Swap 2 numbers using a temporary variable.
11. Write a program to Swap 2 numbers without using a temporary variable.
12. Write a program to convert degrees fahrenheit into Degrees Celsius.
13. Write a program to covert a floating point number into the corresponding integer.

14. Write a program to convert an Integer into the Corresponding floating point number.
15. Write a program to determine whether a person is eligible to vote or not.
16. Write a program to determine the character entered by the user.
17. Write a program to find whether the given number is even or odd.
18. Write a program to find whether the given year is a leap year or not.
19. Write a program to test whether a number entered is positive , negative or equal to zero.
20. Write a program to input three numbers and then find largest of them using && operator.
21. Write a program to calculate the roots of a quadratic equation.
22. Write a program to demonstrate the use of switch statement without a break.
23. Write a program to determine whether an entered character is a vowel or not.
24. Write a program to enter a number from 1 to 7 and display the corresponding day of the week using switch case statement.
25. Write a program that accepts a number from 1 to 10. Print whether the number is even or odd using a switch case construct.

26. Write a program to print the following pattern

```

*****
*****
*****
*****

```

27. Write a program to print the following pattern

```

A
AB
ABC
ABCD
ABCDE

```

28. Write a function to swap the value of two variables.
29. Write a program to calculate area of a circle using function.
30. Write a program to print the Fibonacci series using recursion.
31. Write a program to read and display n numbers using an array.
32. Write a program to print the position of the smallest of a number using arrays.
33. Write a program to enter n number of digits. Form a number using these digits.
34. Write a program to transpose a 3X3 matrix.
35. Write a program to read and display a 3X3 matrix.
36. Write a program to enter n number of digits. Form a number using these digits.

37. Write a program to implement linear search.
38. Write a program to implement Binary search.
39. Write a program to print the elements of a 2D array.
40. Write a program to display a string using printf ().
41. Write a program to find the length of a string.
42. Write a program to convert characters of a string into lower case.
43. Write a program to print Hello worlds using pointers.
44. Write a program to calculate area of circle.
45. Write a program to display the sum and average of numbers from m to n.
46. Write a program to calculate area of a Triangle.
47. Write a program using structures to find the largest of three numbers.
48. Write a program to read a file character by character, and display it simultaneously on the screen.
49. Write a program to count the number of characters and number of lines in a file.
50. Write a program to copy one file into another .copy one character at a time.

**Course Code: BCA120B**

<b>L (Hr.)</b>	<b>T (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
0	0	2	2

**Course Objectives:**

The purpose of this course is to enhance the practical knowledge based on prescribed theory course. The students will be able to enhance their analyzing and problem solving skills after implementation of all the given experiments.

**List of Sample Programs**

1. Create a Web Page using HTML To Display “Welcome to the World of HTML” using basic HTML tags.
2. Create a Web Page using HTML using Headings, Paragraphs and Breaks HTML Tags.
3. Write a program to design a Bio-Data.
4. Write a program in html to create a webpage with four frames (Picture, table, list, and hyperlink)
5. Write a program to create a Ordered List in HTML
6. Write a program to create a Unordered List in HTML
7. Write a program to create a Defined List in HTML
8. Write a program in html to create a webpage to show different hobbies.
9. Write a program in html to Implementation Of Table Tag
10. Write a program in html to create a webpage with four frames (Picture, table, list, and hyperlink).
11. Write a program in html to create a webpage to show the block level elements and text level elements.
12. Write a program in html to create a webpage to show various confectionary items using ordered list and unordered list.
13. Write a program in html to create a webpage to show different hobbies.
14. Write a program in html to show India map.
15. Write a program in html to create a web page using style sheet.
16. Write a program in html to create a web page to show registration naukri.com.

17. Write a program in html to show books in inventory in different tables by using rowspan and colspan.
18. Create a Web Page in HTML to show Admission form in some university.
19. A Web Page in HTML to show all the Text, Color, Background and Font Elements.
20. Write a Program to Create a Nested List.
21. Write a program to create menu using HTML and CSS
22. Design of the cart page and the registration page required for online book store.
23. Create a static webpage using table tags of HTML
24. Create a static web page which defines all text formatting tags of HTML in tabular format.
25. Create webpage using list tags of HTML
26. Create employee registration webpage using HTML form objects
27. Apply style sheet in Web page. [inline, embedded and linked]
28. Create a simple xml file and also create dynamic web page in which XML tags used.

**Course Name: UNIX and Shell Programming Lab**

**Course Code: BCA121A**

<b>L (Hr.)</b>	<b>T (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
0	0	2	2

**Course Objectives:**

The purpose of this course is to enhance the practical knowledge based on prescribed theory course. The students will be able to enhance their analyzing and problem solving skills after implementation of all the given experiments.

**List of Sample Programs**

1. Demonstrate how to use the following Bourne Shell commands: cat, grep, ls, more, ps, chmod, finger, ftp, etc.
2. Use the following Bourne Shell constructs: test, if then, if then else, if thenelif, for, while, until, and case.
3. Copy, move, and delete files and directories.
4. Write moderately complex Shell scripts.
5. Make a Shell script executable.
6. Create a ".profile" script to customize the user environment.
7. Execute programs written in C under UNIX environment
8. Learn tracing mechanisms (for debugging), user variables, Bourne Shell variables, read-only variables, positional parameters, reading input to a Bourne Shell script, command substitution, comments, and exporting variables. In addition, test on numeric values, test on file type, and test on character strings are covered.
9. Use advanced features of File Transfer Protocol (FTP)
10. Compile source code into object and executable modules.

**Semester III**

**Course Name: Principles and Practices of Management**

**Course Code: BCA122A**

<b>L (Hr.)</b>	<b>T (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	0	0	3



## **Course Objectives:**

1. To discuss, understand, and present management principles, processes and procedures in consideration of their effort on individual actions.
2. To participate, summarize and lead class discussions case problems
3. To be able to understand the concept of environment and environment relationship.
4. To knowledge and understanding of the Principles of Management
5. To enable the students to manage valuable insight into the workings of business and other organizations.

## **Syllabus**

### **Unit I**

**Introduction:** Concepts of management, The management process and function, Types and roles of managers, Managerial levels and skills, Emerging challenges for management.

Management Theories: Scientific management school, Administrative management, Behavioral School, Management science school, Systems approach, Contingency approach.

### **Unit II**

**Environmental Context:** Concept of environment, External environment and internal environment, Organization- environment relationships, Managerial ethics, Social responsibility, Planning and Decision Making: Planning an overview, Importance of planning, Purpose and functions of organizational goals, Types of planning, Concept of strategic planning, Situational analysis, Managerial decision making: nature of decision making, styles and conditions of decision making, group decision making, tools for planning and decision making

### **Unit III**

**Organizational Design and Structure:** Meaning, Principles and approaches to organizing, job design, Departmentation, Nature and types of organization design, Situational influence on organizational design, informal organization, Authority, power and responsibility, Delegation and decentralization of authority, Emerging concepts in organizing

### **Unit IV**

**Motivation:** the motivational process, the need-hierarchy and motivation-hygiene theories, Motivation through employee participation

Communication and Team Work: Concept, nature, and forms of organizational communication, Informal communication, Interpersonal communication, barriers to communication, Groups and teams: concept and characteristics, Interpersonal and inter-group conflicts, Managing conflicts

**Unit V**

**Control and Quality Management:** Elements of control, Nature and types of control, Managing control in organization, Information for effective control, MIS and DSS, TQM, Factors effecting quality. Deming management

International Management: Globalization – concept, nature and forms. Methods of globalization, changing international management scenario. Multinational companies – meaning and types, effects and benefits

**Course Outcomes (COs):**

**Upon successful completion of this subject students should be able to:**

CO1: Introduction to relation between management and behavior.

CO2: Define management, quality management, and project management in respect of planning & Decision making.

CO3: Identify relevant issues in human resource management with leadership qualities.

CO4: Understands the theories and practices of Communication and team work in management.

CO5: Describe International Management with Quality control concepts.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

Course Outcome	Program Outcome											
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H							L				H
CO2		M				H	M					
CO3				H						H		
CO4							H					
CO5	H				M						L	

H = Highly Related; M = Medium; L = Low

### **Text Books**

1. Stoner J.A. and Freeman R.E, “Management”, VI Edition , Prentice hall India
2. Parkinson C.N., Rustomji M.K. and Sapre S.A, “Great Ideas in Management”.

### **Reference Books**

1. S.K. Mandal , “Management: Principles and Practice”
2. Lallan Prasad, S S Gulshan, “Management Principles and Practices”.
3. “Great Ideas in Management” by Parkinson C N and Rustomji M K and Sapre S A

**Course Name: Programming in PHP**

**Course Code: BCA123A**

<b>L (Hr.)</b>	<b>T (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	0	0	3

### **Course Objectives:**

1. To demonstrate the basics of the PHP and MYSQL, PHP configuration in IIS & Apache Web Server and features of PHP.
2. To understand how PHP, HTML and MYSQL work together to produce dynamic pages.
3. To apply specific PHP variables, data types, syntax and conditional statements.
4. To be able to apply control statements and database connectivity.
5. To make use of string functions and array functions.

## **Syllabus**

### **Unit I**

Introduction to PHP, Installation of PHP and MySQL, PHP configuration in IIS & Apache Web Server and features of PHP, Writing PHP, How PHP code is parsed, Embedding PHP and HTML Executing PHP and viewing in Browser.

### **Unit II**

Data types, Operators, PHP variables: static and global variables, Comments in PHP, Control Structures, Condition statements, If...Else, Switch, ? operator, Loops, While, Break Statement Continue. Do...While, For, For each, Exit, Die, Return

### **Unit III**

Arrays in PHP, Working With Data, FORM element, INPUT elements, Validating the user input, Passing variables between pages, Passing variables through GET, Passing variables through POST, Passing variables through REQUEST, Working With Data, Built-in functions.

### **Unit IV**

String.Functions: chr, ord, strtolower, strtoupper, strlen, ltrim, rtrim, substr, strcmp, strcasecmp, strpos, strrpos, strstr, stristr, str\_replace, strrev, echo, print, Math Functions: abs, ceil, floor, round, fmod, min, max, pow, sqrt, rand.

### **Unit V**

Array Functions: count, list, in\_array, current, next, previous, end, each, sort, rsort, array\_merge, array\_reverse User Defined Functions. Sessions and cookies, Concept of Session Starting session, Modifying session variables, registering and deleting session variable Concept of Cookies.

### **Course Outcomes:**

#### **Upon successful completion of this subject students should be able to:**

CO1: Demonstrate the basics of the PHP and MYSQL, PHP configuration in IIS & Apache Web Server and features of PHP.

CO2: Examine how web pages are developed using PHP array, form and input elements.

CO3: Apply specific PHP variables, data types, syntax and conditional statements.

CO4: Understand the basics of array functions and user defined functions.

CO5: Understand the basics of string functions like chr, ord, strtolower, strtoupper, strlen.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

Course Outcome	Program Outcome											
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H							L				H
CO2		M				H	M					
CO3				H						H		
CO4							H					
CO5	H				M						L	

H = Highly Related; M = Medium; L = Low

**Text Books**

1. PHP Cookbook by David Sklar and Adam Trachtenberg, O'Reilly Media, Inc., ISBN: 978-1-449-36375-8.
2. Core PHP Programming, Leon Atkinson Pearson publishers, 2nd Edition, ISBN-13: 978-0130893987.

**Reference Books**

1. The Complete Reference PHP, SteverHolzner McGraw Hill, Edition: 1 ISBN: 9780070223622.
2. Web Technologies: HTML, JAVASCRIPT, PHP, JAVA, JSP, ASP.NET, XML and Ajax, Black Book: HTML, Javascript, PHP, Java, Jsp, XML and Ajax, Black Book” by Kogent Learning Solutions Inc.
3. “Web Enabled Commercial Application Development Using HTML, JavaScript, DHTML and PHP ( 4th Revised Edition ) CD-ROM Included” by Ivan Batross

**Course Name: Data Structures and Algorithms using C**  
**Course Code: BCA125A**

<b>L (Hr.)</b>	<b>T (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	1	0	4

**Course Objectives:**

1. To impart the basic concepts of data structures and algorithms.
2. To understand concepts about searching and sorting techniques.
3. To understand basic concepts about stacks, queues, lists, trees and graphs.
4. To understanding about writing algorithms and step by step approach in solving problems with the help of fundamental data structures.
5. To impart the basic concepts of algorithms implementation in optimized time .

**Syllabus**

**Unit I**

**Introduction to Data Structures and Algorithms:** Elementary Data Structure Organization, Classification of Data Structures, Operation of Data Structures, Operations on Data Structures ,Abstract Data Type ,Algorithms, Different Approaches to, Designing an Algorithm,

Control Structures Used in Algorithms, Time and Space Complexity, Omega Notation ( $\Omega$ ), Theta Notation ( $\Theta$ ), Other Useful Notations.

## UNIT II

**Array and Linked List:** Declaration of Arrays, Accessing the Elements of an Array, Storing Values in Array, operations, Passing Array to functions, Pointers and Arrays, Arrays of Pointers, Two-dimensional Arrays, Operations on , Passing Two-dimensional Arrays to Functions, Pointers and Two-dimensional Arrays, Sparse Matrices .

Linked list Basic Terminologies, Memory Allocation and De-allocation for a Linked List, Singly Linked Lists, Circular Linked Lists, Doubly Linked Lists, Circular Doubly Linked Lists, Header Linked Lists, Multi-linked Lists, Applications of Linked Lists.

## UNIT III

**Stack:** Array Representation of Stack, Operations on Stack, linked Representation of Stacks, Operations on a Linked Stack, Multiple Stacks, Applications of Stack,

**Queues:** Introduction to Queues, Array Representation of Queues ,Linked Representation of Queues, Types of Queues , Applications of Queues

## UNIT IV

**Trees and BST** Trees: Types of trees , Creating a Binary Tree from a General Tree, Traversing a Binary Tree, Huffman's Tree

Binary Search Trees: BST Operations, Threaded Binary Trees, AVL Trees, Red-Black Trees, Splay Trees

## UNIT V

**Graph, Searching & Sorting:** Basic Terminologies, Directed Graphs, Representations of Graphs, Graph Traversals Algorithms, Topological Sorting, Shortest-Path Algorithms.

**Searching & Sorting:** Introduction to searching, Linear and Binary Search, Interpolation Search, jump search, Sorting Types, Bubble, Insertion, Selection , Merge Sort, Radix Sort Shell Sort, Quick Sort, Heap Sort.

**Course Outcomes (COs):**

**Upon successful completion of this subject students will able**

CO1: Ability to analyse algorithms and a algorithm correctness.

CO2: Ability to implement various techniques of link list.

CO3: Ability to describe stack, queue with linked list operation.

CO4: Ability to have knowledge of tree and graphs concepts.

CO5: Ability to summarize searching and sorting techniques

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

Course Outcome	Program Outcome											
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1		H							M			
CO2						M				H		
CO3				H				M				
CO4						M					H	
CO5	H				M							L

H = Highly Related; M = Medium; L = Low

**Text Books**

1. R. G. Dromey, "How to Solve it by Computer", Second Edition, Prentice-Hall of India, 2002.
2. Reema Thereja,, "Data Structure using C" , Second Edition , Oxford University Press, 2014

**Reference Books**

1. M. A. Weiss, "Data Structures and Algorithm Analysis in C", Second Edition, Pearson Education Asia, 2002.
2. ISRD Group, "Data Structures using C", Fifth Edition Tata McGraw Hill, 2007
3. Richard F. Gilberg, Behrouz A. Forouzan, "Data Structures – A Pseudocode" Third Edition Prentice-Hall of India, 2004.



**Course Name: Programming Fundamentals using C++**

**Course Code: BCA126A**

<b>L (Hr.)</b>	<b>T (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	1	0	4

**Course Objectives:**

1. To explain the difference between object oriented programming and procedural programming and features of object oriented programming.
2. To be able to create programs using more advanced C++ features such as composition of objects, operator overloads, dynamic memory allocation, inheritance and polymorphism, file I/O, exception handling, etc.
3. To be able to build C++ classes using appropriate encapsulation, objects and functions.
4. To be able to apply operator overloading in different form.
5. To be able to understand the working of files.

**Syllabus**

**Unit I**

**Introduction to Object Oriented Programming:** Generation of Programming, Programming Paradigms, Features of Object Oriented Programming , Introduction to C++, History of C++, Structure, First Program, Compiling and Executing C++, Using Comments, Tokens, Character Set, Keywords, Identifiers, Data Types, Variables, Constants, Input and Output Statements, Operators in C++, Operator Precedence and Associability, Decision Control and Looping, Statements, Functions, Arrays, Strings, Pointers, Structure, Union, and Enumerated Data Types

**Unit II**

**Classes and Objects:** Introduction , Class, Creating Objects, Accessing Object Members, Nested Member Functions, Making a Member Function Inline, Memory Allocation for Class and Objects, Array of Objects, DMA, Objects as Function Arguments, Returning Objects, this pointer, Constant

Parameters and Members, Pointers Within a Class , Empty Classes, Friend Classes, Constructor, Types of Constructors, Constructor with Arguments, Constructor Overloading, Destructors

### **Unit III**

**Operator Overloading and Type Conversions:** Scope of Operator Overloading, Syntax, Not Overloading Operators, Implementing Operator, Overloading Unary Operators, Overloading Binary Operators, Overloading Special Operators, Type Conversions

### **Unit IV**

**Inheritance:** Defining Derived Classes, Access Specifiers, Inheritance, Types of Inheritance, Single Inheritance, Constructors and Destructors in Derived Class, Constructor in Multi-Level, Multi-Level Inheritance, Constructor in Multi-Level Inheritance, Multiple Inheritance, Hierarchical Inheritance, Multi-path Inheritance.

### **Unit V**

**File Handling:** Streams in C++, Classes for File Stream, Opening and Closing of Files, Detecting the End-of-File, Files Modes, File pointer.

### **Course Outcomes:**

**Upon successful completion of this subject students should be able to:**

CO1: Understand the features of C++ supporting object oriented programming

CO2: Be able to program using more advanced C++ features such as composition of objects, operator overloads, dynamic memory allocation, inheritance and polymorphism, file I/O, exception handling, etc.

CO3: Be able to apply operator overloading in different form.

CO4: Understand how to apply the major object-oriented concepts to implement object oriented programs in C++, encapsulation, inheritance and polymorphism

CO5: Able to understand the working with files like opening and closing of files.

### **MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

Course	Program Outcome
--------	-----------------

Outcome												
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1		L	H				M		M			
CO2		H			L			M				L
CO3	H				H	M			L			
CO4			H	H			L	M				
CO5		L							H		M	

H = Highly Related; M = Medium; L = Low

### Text Books

1. Reema Thereja, "Object Oriented Programming with C++", First Edition, Oxford University Press ,2015
2. Let Us C: Bala Guruswamy, TATA McGraw Hill.

### Reference Books

1. Object Oriented Programming with C++, Sourav sahay, Oxford University, Sept 2012.
2. Richard F. Gilberg, Behrouz A. Forouzan, "Data structures, A Pseudocode Approach with C", (2e), Cengage Learning India Pvt.Ltd, India, 2009.
3. Robert Kruc & Bruce Lening, "Data structures & Program Design in C", (2e), Pearson, 2007.

**Course Name: PHP Lab**

**Course Code: BCA128A**

L (Hr.)	T/P (Hr.)	Pr (Hr.)	Credits
0	0	2	2

## Course Objectives

The purpose of this course is to enhance the practical knowledge based on prescribed theory course. The students will be able to enhance their analyzing and problem solving skills after implementation of all the given experiments.

## List of Sample Programs

1. WAP to reverse the string without using any predefined function
2. WAP to check the season of the year according to given month using control statement (Switch statement)
  - a. spring month 3,4,5
  - b. summer month 6,7,8
  - c. autumn month 9,10,11
  - d. winter month 12,1,2
3. WAP to create an array for employee names and associate another array of employee's salary and print these values using for each loop.
4. WAP that will accept an array of integers as input and output an array where for each item in the source array, the new array will perform the following operations:-
  - a. for even numbers divide by 2
  - b. for odd number multiply by 3
5. WAP to make sure a value has been supplied for a form element. For example, you want to make sure a text box hasn't been left blank.
6. WAP to know whether an email address a user has provided is valid.
7. WAP in php to show the values in the input fields after the user hits the submit button
8. WAP create a form with one text field and submit button to find string in lower case, uppercase, sub string, position of a string and trimmed string
9. WAP to find whether a given string is subset of another given string or not
10. WAP to find square and square root of a randomly generated number
11. WAP to merge 2 associative array sort that array in :-
  - a. ascending order
  - b. descending order
  - c. in key sorted form
12. WAP to generate captcha code
13. WAP to create session, provide it name and id and display their values and finally delete the session.

14. WAP to in which you want to set a cookie so that your website can recognize subsequent requests from the same web browser.
15. WAP to in which you want to read the value of a cookie that you've previously set.
16. WAP in which You want direct access to the body of a request, not just the parsed data that PHP puts in \$\_POST for you. For example, you want to handle an XML document that's been posted as part of a web services request.
17. WAP to shows how to use validate().
18. WAP to Call ob\_start() at the top of your page and ob\_end\_flush() at the bottom. You can then intermix commands that generate output and commands that send headers.
19. WAP to send compressed content to browsers that support automatic decompression.
20. WAP to use the same HTML page to emit a form and then process the data entered into it. In other words, you're trying to avoid a proliferation of pages that each handle different steps in a transaction.
21. WAP in which you have data that can be easily represented as key/value pairs, want to store it safely, and have very fast lookups based on those keys.
22. WAP in which you want access to a SQL database to store or retrieve information. Without a database, dynamic websites aren't very dynamic.
23. WAP in which you want to retrieve some data from your database.
24. WAP in which you want a concise way to execute a query and retrieve the data it returns.
25. WAP in which you want to know how many rows a SELECT query returned, or you want to know how many rows an INSERT, UPDATE, or DELETE query changed.
26. WAP in which you need to make text or binary data safe for queries

**Course Name: Data Structure and Algorithm using C Lab**

**Course Code: BCA129A**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
0	0	2	2

### **Course Objectives**

The purpose of this course is to enhance the practical knowledge based on prescribed theory course. The students will be able to enhance their analyzing and problem solving skills after implementation of all the given experiments.

### **List of Sample Programs**

1. Write a menu driven program to implement the following sparse matrices using one-dimensional array:
  - i) Diagonal Matrix
  - ii) Lower Triangular Matrix
  - iii) Upper Triangular Matrix
  - iv) Symmetric Matrix
2. Write a program to compute  $b^r$  using recursion where b represent base and r represents power.
3. Write a program to reverse a user entered string using recursion.
4. Write a program to perform the following Queue operations using Circular Array implementation (Use Templates):
  - i) Enqueue
  - ii) Dequeue
5. Write a program to add two large integers using stack.
6. Write a program to evaluate postfix expression using stack.
7. Write a program to implement Linked List using templates. Include functions for insertion, deletion and search of a number, reverse the list and concatenate two linked lists (include a function and also overload operator +).
8. Write a program to perform the following Stack operations using linked list.
  - i) Push
  - ii) Pop
  - iii) Clear
9. Write a program to create and perform the following operations on Queues using linked list:
  - i) Enqueue
  - ii) Dequeue
10. Write a program to implement Doubly Linked List using templates. Include functions for insertion, deletion and search of a number, reverse the list.
11. Write a program to implement Circular Linked List using templates. Include functions for insertion, deletion and search of a number, reverse the list.
12. Write a program to add two polynomials using linked list representation.
13. Write a menu driven program to implement the insertion operations in an ordered linked list.
14. Write a menu driven program to implement the deletion operations in an ordered linked list.
15. Write a menu driven program to implement the merging operations in an ordered linked list.
16. Write a Program to reverse elements of a Stack using an additional Stack.
17. Write a Program to reverse elements of a Stack using an additional Queue.

18. Write a Program to implement the insertion operation in a Binary Search Tree.
19. Write a Program to implement the deletion by coping or by merging operation in a Binary Search Tree.
20. Write a Program to implement the Search a number in BST operation in a Binary Search Tree.
21. Write a Program to implement the count leaf nodes of the tree operation in a Binary Search Tree.
22. Write a menu driven program to implement the Selection sorting algorithms.
23. Write a menu driven program to implement the Insertion sorting algorithms.
24. Write a menu driven program to implement the Binary Search algorithms.
25. Write a menu driven program to implement the Bubble sorting algorithms.

**Course Name: Programming Fundamentals using C++ Lab**

**Course Code: BCA130A**

<b>L (Hr.)</b>	<b>T (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
0	0	2	2

**Course Objectives:**

The purpose of this course is to enhance the practical knowledge based on prescribed theory course. The students will be able to enhance their analyzing and problem solving skills after implementation of all the given experiments.

**List of Sample Programs**

1. Write a program to read and print values of variables of different data types.
2. Write a program to calculate the distance between two points.
3. Write a program to perform addition, subtraction, multiplication and division on 2 integer numbers.
4. Write a program to find largest of two numbers using ternary operator.
5. Write a program to print the ASCII value of a corrector.
6. Write a program to swap 2 numbers using a temporary variable.

7. Write a program that displays the size of every data type.
8. Write a program to convert an integer into the corresponding floating point number.
9. Write a program to enter any character. If the entered character is in lower case, convert it into upper case. If it is a lower case character, then convert it into upper case.
10. Write a program to find whether a given year is leap year or not.
11. Write a program to print 20 horizontal asterisks (\*)
12. Write a program to display the largest of 10 numbers using ternary operator.
13. Write a program to print the following pattern:

```
A
AB
ABC
ABCD
ABCDE
```

14. Write a program to print the following pattern:

```
1
121
12321
1234321
```

15. Write a program to classify a given number as prime or composite.
16. Write a program using for loop to calculate the average of first n natural numbers.
17. Write a program to enter a decimal number. Calculate and display the binary equivalent of this number.
18. Write a program to sum the series:  $1/1+2^2/2+3^3/3+\dots$
19. Write a program to generate calendar of a month given the start day and the number of days in that month.
20. Write a program to calculate the area of a circle using functions.
21. Write a program to calculate the volume of a cuboid using default arguments.
22. Write a program to add two values of different data types using static polymorphism.
23. Write a program to calculate GCD using recursive function.
24. Write a program to calculate  $\exp(x,y)$  using recursive functions.
25. Write a program to read and display n random numbers using an array.
26. Write a program to interchange the largest and the smallest number in the array.
27. Write a program to find the second largest number using an array of n numbers.
28. Write a program to merge 2 unsorted arrays.



29. Write a program to implement linear search in any given array.
30. Write a program to transpose a 3\*3 matrix.
31. Write a program to read and display a 2\*2\*2 array.
32. Write a program to concatenate 2 strings.
33. Write a program to compare 2 strings.
34. Write a program to insert a string in the main text.
35. Write a program to sort names of students.
36. Write a program to enter a text that contains multiple lines. Display the n lines of text starting from m<sup>th</sup> line.
37. Write a program to add 2 integers using pointer.
38. Write a program to add 2 vectors (Arrays).
39. Write a program to copy a given string into a new string. Memory for the new string must be allocated dynamically.
40. Write a program to read, display, add and subtract 2 complex numbers.
41. Write a program to read and display information of a student using structure within a structure.
42. Write a program to read and display information of all the students in the class.
43. Write a program using pointer to structure to initialize the members in the structure.
44. Write a program to illustrate the use of arrays within structure.
45. Write a program to display the name of the colors using enumerated types.
46. Write a program to enter a rational number, simplify and display it.
47. Write a program to demonstrate the concept function overloading in the member functions of a class.
48. Write a program that adds 2 complex numbers. The object must be passed through reference and the result must be passed by value.
49. Write a program to display the list of students according to their marks.
50. Write a program to find mean of 2 numbers belonging to two different classes using friend function.

**Course Name: Multimedia Lab (Audio/Video editing tools, Flash, Dreamweaver)**

**Course Code: BCA167A**

<b>L (Hr.)</b>	<b>T (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
0	0	2	2

**Course Objectives:**

The purpose of this course is to enhance the practical knowledge based on prescribed theory course. The students will be able to enhance their analyzing and problem solving skills after implementation of all the given experiments.

**List of Experiment**

1. Video Production/Editing  
Macs – iMovie (Apple iLife Suite)  
Windows – Corel VideoStudio Ultimate
2. Sound Production/Editing  
Macs – GarageBand (Apple iLife Suite)  
All – Audacity + LAME encoder
3. Scanning  
All – Epson Scan  
Macs – Apple Scan  
Windows – Windows Fax & Scan
4. Photo (bitmap) Editing  
All – Affinity Photo  
All – Gimp
5. Drawing (vector) Production/Editing  
All- Affinity Designer

All – Inkscape

6. Productivity/Office

All – LibreOffice

All – GoogleApps

7. PDF Production/Editing

All – LibreOffice

8. Web Site/Page Production

All – GoogleSites

All- WordPress

# Semester IV

**Course Name: Programming in Java**

**Course Code: BCA133A**

<b>L (Hr.)</b>	<b>T (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	1	0	4

## **Course Objectives:**

1. To be able to understanding the concept of programming paradigms, basic concept of object oriented programming and features of java.
2. To demonstrate the concept of data types, literals and basic structure of java programming.
3. To enhance the practical knowledge of decision making statements and control statements.
4. To be able to use variety technologies of Java and work with different platforms.
5. To examine the life cycle of applets and packages.

## **Syllabus**

### **Unit I**

**Introduction to OOPS:** Paradigms of Programming Languages – Basic concepts of Object Oriented Programming – Differences between Procedure, Oriented Programming and Object Oriented , History of Java features – Java Environment, JDK ,API.

Introduction to Java: Types of java program, Creating and Executing a Java program, Java Tokens: Keywords, Character set, Identifiers, Literals, Separator, Java Virtual Machine (JVM) – Command Line Arguments, Comments in Java program.

### **Unit II**

**Control Structures, Arrays, and Vectors:** Elements, Constants, Variables, Data types, Scope of variables, Type casting – Operators: Special operators, Expressions, Evaluation of Expressions.

Decision making and Branching: Simple if statement, if – else statement, nesting if – else, else if Ladder – switch statement – Decision making and looping: While loop, do - While loop for loop, break, loop continue Statement.

### **Unit III**



CO1	H		M			L				M		
CO2		H										M
CO3			H								M	
CO4					M			H				
CO5	H			M					M			

H = Highly Related; M = Medium; L = Low

### Text Books

1. Programming in Java, Sachin Malhotra, Oxford University, November 2013
2. Java One step ahead, Seth and Juneja, Oxford University, May 2017.

### Reference Books

1. Programming with Java A Primer, E. Balaguruswamy Tata McGraw Hill Companies
2. R. Nageswara Rao, "Core Java: An Integrated Approach", First Edition, DT Editorial Services, 2016.
3. Herbert Schildt, "The Complete Reference", Ninth Edition, McGraw Hill, 2014.
4. Cay S. Horstmann, "Core Java", Ninth Edition, Prentice Hall, 2012.

**Course Name: Programming in Python**

**Course Code: BCA134A**

L (Hr.)	T (Hr.)	Pr (Hr.)	Credits
3	0	0	3

### Course Objectives:

1. To lineup the basics of python programming, Features, history, data types and variables.
2. To be able to solve real-world problems through python programming.
3. To be able to understand problem solving approaches, programming languages, object oriented programming.
4. To construct of Python language such as control statements, functions, strings, files, data structures.
5. To apply the concept of Classes and objects, functions and array in python.

## **Syllabus**

### **Unit I**

**Basics Of Python Programming:** Features of Python ,History of Python ,The Future of Python ,Writing and Executing First Python Program, Literal Constants , Numbers , Strings, Variables and Identifiers , Data Types , Input Operation, Comments , Reserved Words , Indentation.

### **Unit II**

Operators and Expressions, Expressions in Python, Operations on Strings , Concatenation, Multiplication (or String Repetition), Other Data Types, Tuples , Lists , Dictionary, Type Conversion .

### **Unit III**

**Decision Control, Function and Modules:** Introduction to Decision Control Statements, Selection/Conditional Branching Statements, Basic Loop Structures/ Iterative Statements, while loop, Nested Loops, The break Statement, The continue Statement, The pass Statement, The else Statement used with Loops.

### **Unit IV**

Function: Function Definition, Function Call, Variable Scope and Lifetime, return statement, Lambda Functions or Anonymous Functions , Documentation Strings, Recursive Functions.

### **UNIT V**

**File Handling** File : File Path, Types of Files, Opening and Closing Files, Reading and Writing Files, File Positions Renaming and Deleting Files, Directory Methods

### **Course Outcomes (COs):**

**Upon successful completion of this subject students should be able to:**

CO1: Outline the basics of python programming, Features, history, data types and variables.

CO2: Understanding of Python especially Decision control, function and modules.

CO3: Understanding of the file handling and Data Structure.

CO4: Be exposed to object oriented concepts in Python programming, decision controls and function.

CO5: Understand the concept of file, opening, closing, reading and writing files.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

Course Outcome	Program Outcome											
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	H								M			
CO2	H		M									M
CO3				M						M		
CO4							H					
CO5		M			M			H				

H = Highly Related; M = Medium; L = Low

**Text Books**

1. Python Programming, Oxford, ReemaThareja, June 2017
2. “Programming Python” by Mark Lutz and O’Reilly Media

**Reference Books**

1. “Python Testing Cookbook” by Greg L Turnquist
2. “Head First Programming” by Paul Barry and David Griffiths
3. “Python Crash Course: A Hands-On, Project-Based Introduction to Programming” by Eric Matthes.



**Course Name: Computer Network**

**Course Code: BCA135A**

<b>L (Hr.)</b>	<b>T (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	0	0	3

**Course Objectives:**

1. Explain the importance of data communications and the Internet in supporting business communications and daily activities.
2. Explain various switching techniques and functions of physical layer.
3. Recognize the working of different internetworking devices and their functions.
4. Explain the role of protocols in networking and analyze the services and features of the network layer of data networks.
5. Analyze the working of transport layer protocols and compare TCP and UDP.

## **Syllabus**

### **Unit I**

**Introduction:** Network definition, Network topologies, Network classifications, Layered network architecture, Overview of ISO-OSI reference model, Overview of TCP/IP protocol suite. Multiplexing techniques, Transmission media.

### **Unit II**

Networks Switching Techniques and Access mechanisms: Circuit switching; Packet switching- Connectionless datagram switching, Connection-oriented virtual circuit switching. Data Link Layer Functions and Protocol: Error detection and error correction techniques, Data-link control- framing and flow control, Error recovery protocols- Stop and wait ARQ, Go-back-n ARQ.

### **Unit III**

Multiple Access Protocol and Networks: LANS, Ethernet, Token Ring, WAN, WAN Architecture, ALOHA, CSMA/CD protocols, Repeaters, Hubs, Switches, Bridges, Router and Gateways.

### **Unit IV**

Networks Layer Functions and Protocols: Routing, Routing algorithms, Network layer protocol of Internet- TCP/IP protocol, IP Addressing, IP Datagram Packet. Internet control protocols – ICMP, ARP, RARP.

### **Unit V**

Transport Layer Functions and Protocols: Transport services, Transport layer protocol of Internet- UDP and TCP. Relationship between TCP and IP, TCP Connections, TCP Packet format.

### **Course Outcomes (Cos):**

#### **Upon successful completion of this subject students will able**

CO1: Independently understand basic computer network and models.

CO2: Understand and explain Data Communications System, switching, error correction and detection.

CO 3: Identify the different types of network devices and protocols.

CO 4: Enumerate the network layer functions and protocols.

CO 5: Understand transport layer functions and protocols.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

Course Outcome	Program Outcome											
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1				H					M			
CO2		M		M						M		
CO3	M				M		L					M
CO4			L					M				
CO5	H				L				M			

H = Highly Related; M = Medium; L = Low

**Text Books**

1. A.S Godbole , Data Communications and Networks” , 2<sup>nd</sup> Edition, Tata Mc Graw Hill,2014
2. B. A. Forouzan: Data Communications and Networking, 4<sup>th</sup> edition, Tata McGraw Hill Education Private Ltd., 2007.

**Reference Books**

1. A. S. Tanenbaum: Computer Networks, 4<sup>th</sup> edition, PEARSON, 2003.
2. D. E. Comer, M.S.Narayanan, Computer Networks and Internets with applications, Fourth Edition, PEARSON ,2008.

**Course Name: Data Science & Analytics**

**Course Code: BCA169A**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
2	1	0	3

**Course Objectives**

1. Able to explain Statistical Analysis, Python Environment Setup and working of Numpy.
2. Able to design Data Visualization in Python using matplotlib.
3. Able to perform linear and multiple linear regression analyses.
4. Ability to select and implement machine learning techniques in real life applications.
5. Learn about computing environment that is suitable for the applications under consideration.
- 6.

**Syllabus**

**Unit - I**

Data Science Overview, Data Analytics Overview, Statistical Analysis and Business Applications, Python Environment Setup and Essentials, Mathematical Computing with Python – NumPy: - Introduction to Numpy, Creating and Printing an ndarray, Class and Attributes of ndarray, Basic Operations, Slicing, Mathematical Functions of Numpy.

**Unit- II**

Data Manipulation with Pandas: - Introduction to Pandas, Understanding DataFrame, View and Select Data, Missing Values, Data Operations, File Read and Write Support, Pandas Sql Operation, Analyse different Dataset using Pandas.

**Unit- III**

Data Visualization in Python using matplotlib: - Introduction to Data Visualization, Line Properties, Plot and Subplots, Types of Plots, Draw a pair plot using seaborn library.

#### Unit- IV

Scientific computing with Python (Scipy):- Introduction to SciPy, SciPy Sub Package - Integration and Optimization, SciPy sub package - Statistics, Weave and IO, Solving Linear Algebra problem using SciPy.

#### Unit- V

Machine Learning with Scikit-Learn: Machine Learning Approach, Supervised Learning Model Considerations, Supervised Learning Models - Linear Regression, Logistic Regression, K Nearest Neighbours, Decision Tree: Regression and Classification Trees, Support Vector Machines, Unsupervised Learning Models.

#### Course Outcomes (Cos):

CO1: Students will be able to explain Statistical Analysis, Python Environment Setup and working of Numpy.

CO2: Students will be able to Manipulate and analysis data with Pandas

CO3: Students will be able to design Data Visualization in Python using matplotlib.

CO4: Students will be able to learn about scientific computing with Python

CO5: Students will be able to learn Machine Learning Approaches using Scikit Learn.

#### MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome											
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1				H					M			
CO2		M		M						M		
CO3	M				M		L					M
CO4			L					M				
CO5	H				L				M			

H = Highly Related; M = Medium; L = Low

**Text Books**

1. Hastie, Trevor, et al., The elements of statistical learning. Vol. 2. No. 1. New York: springer, 2009.
2. Montgomery, Douglas C., and George C. Runger., Applied statistics and probability for engineers. John Wiley & Sons, 2010

**Course Name: Programming in Java Lab**

**Course Code: BCA136A**

<b>L (Hr.)</b>	<b>T (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
0	0	2	2

### **Course Objectives:**

The purpose of this course is to enhance the practical knowledge based on prescribed theory course. The students will be able to enhance their analyzing and problem solving skills after implementation of all the given experiments.

### **List of Sample Programs**

1. To write a java program to find the area of rectangle
2. To write a java program to print the individual digits of a 3 digit number.
3. Write a java program to read two integers and print the larger number followed by the words “is larger “If the numbers are equal print the message “These numbers are equal”.
4. To write a java program to read an integer and find whether the number is odd or even
5. To write a java program find the biggest of three integers.
6. Write a program to display a simple message.
7. Write a program to add two numbers.
8. Write a program to arithmetic operations.
9. Write a program to area of the circle.
10. Write a program to check odd or even.
11. Write a program to find Greatest among three numbers.
12. Write a program to find greatest number using nested if.
13. Write a program to sum and average of five marks.
14. Write a program to describe functionality of Arithmetic operations using switch.
15. Write a program to write names of 10 natural numbers.
16. Write a program to sum of ‘n’ natural numbers.
17. Write a program to find Factorial of a number.
18. Write a program to sum of ‘n’ numbers.
19. Write a program to sum of two numbers using class and object.
20. Write a program to prepare mark sheet of a student.
21. Write a program to find area of rectangle using constructor.
22. Write a program to find area of room.
23. To write a java program to find the sum of digits of a given number

24. To write a java program to find the first 15 terms of Fibonacci sequence.
25. To write a java program to print the Armstrong numbers.
26. To write a java program to find the largest and smallest number in an array.
27. Write a program to arithmetic operations using static members.
28. Write a program to print greatest value using nesting of methods.
29. To write a java program that creates a string object and initializes it with your name and performs the following operations
30. To find the length of the string object using appropriate String method.
31. To find whether the character 'a' is present in the string. If yes find the number of times 'a' appear in the name and the location where it appears
32. To write a java program to create a StringBuffer object and illustrate how to append characters and to display the capacity and length of the string buffer
33. To write a java program to create a StringBuffer object and illustrate how to insert characters at the beginning
34. To write a java program to Create a StringBuffer object and illustrate the operations of the append () and reverse () methods.
35. To write a java program to display total marks of 5 students using student class. Given the following attributes: Regno(int), Name(string), Marks in subjects(Integer Array), Total (int).
36. To write a program in java with a class Rectangle with the data fields width, length, area and colour. The length, width and area are of double type and colour is of string type. The methods are get\_length(), get\_width(), get\_colour() and find\_area().
37. Create two objects of Rectangle and compare their area and colour. If the area and colour both are the same for the objects then display "Matching Rectangles", otherwise display "Non-matching Rectangle".
38. Write a java program to create a Player class and inherit three classes Cricket\_Player, Football\_Palyer and Hockey\_Player.
39. Write a program to find area and volume of a room using inheritance .
40. Write a program to describe Method overriding
41. To write a java program to show how a class implements two interfaces.
42. To write a java program to show that the variables in an interface are implicitly static and final and methods are automatically public
43. To write a java program to create a package for Book details giving Book name, Author name, price and year of publishing.



44. To write a java applet program to change the color of a rectangle using scroll bars to change the value of red, green and blue
45. To write an applet program for creating a simple calculator to perform Addition, subtraction, Multiplication and Division using Button, Label and TextField component.
46. To write a java program to catch more than two exception
47. To write a java program to create our exception subclass that throws exception if the sum of two integers is greater than 99.
48. To write a java program for generating two threads, one for generating even number and one for generating odd number.

**Course Name: Programming in Python Lab**

**Course Code: BCA137A**

<b>L (Hr.)</b>	<b>T (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
0	0	2	2

**Course Objectives:**

The purpose of this course is to enhance the practical knowledge based on prescribed theory course. The students will be able to enhance their analyzing and problem solving skills after implementation of all the given experiments.

## List of Experiment

1. Write a program to display data of different types using variables and literals constants.
2. Write a program to reassign values to a variable.
3. Write a program to read variables from the user.
4. Write a program to exhibit indentation errors.
5. Write a program to enter a number and display its hex and octal equivalent and its square root.
6. Write a program to read and print values of variables of different data types.
7. Write a program to calculate area of triangle using Heron's formula.
8. Write a program to calculate the distance between two points.
9. Write a program to perform addition, subtraction, division and multiplication on two floating point numbers.
10. Write a program to perform addition, subtraction, division and multiplication on two integer point numbers.
11. Write a program to calculate average of two numbers. Print their deviation.
12. Write a program to calculate the total amount of money in the piggy bank given the coins of Rs 10, 5, 2, 1.
13. Write a program to convert degrees Fahrenheit into degrees Celsius.
14. Write a program to count all the prime and composite numbers entered by the user.
15. Write a program to find the greatest number from 3 numbers.
16. Write a program to take input from the user and then check whether it is a number or a character.
17. Write a program to separate two values printed on the same line using a tab.
18. Write a program to calculate the sum and average of first 10 numbers.
19. Write a program to find whether the given number is an Armstrong number or not.
20. Write a program to enter a number and then calculate the sum of its digits.
21. Write a program to enter a binary number and convert it into decimal number.
22. Write a program to calculate GCD of 2 numbers.
23. Write a program to print the reverse of a number.
24. Write a program to print the multiplication table of n, where n is entered by the user.
25. Write a program using for loop to calculate the average of first n natural numbers.
26. Write a program using for loop to calculate factorial of a number.
27. Write a program to classified a given number as prime or composite.
28. Write a program to sum the series---  $1+1/2+\dots+1/n$ .

29. Write a program using while loop to read the numbers until -1 is encountered. Also count the numbers of prime numbers and composite numbers entered by the user.
30. Write a program to demonstrate the continue statement.
31. Write a program to write a function that displays a string repeatedly.
32. Write a program to demonstrate the mismatch between function parameters and arguments.
33. Write a program to demonstrate the use global statement.
34. Write a program to demonstrate name clash of local and global variable.
35. Write a program to demonstrate access of variables in inner and outer functions.
36. Write a program to demonstrate flow of control after the return statement.
37. Write a program to write another function which returns an integer to the caller.
38. Write a program that adds two numbers using the syntax of lambda functions.
39. Write a program to use a lambda function with an ordinary function.
40. Write a program to add two numbers using lambda function.

**Course Name: Minor Project**

**Course Code: BCA139A**

<b>L (Hr.)</b>	<b>T (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
0	0	4	4

**Course Objectives:**

The purpose of minor project is to enhance the practical knowledge based on prescribed languages which are introduced in this program. The students will be able to enhance their analyzing and problem solving skills after implementation of all the given experiments.

## **Semester V**

**Course Name: Introduction to Accounting and Economics**

**Course Code: BCA132A**

<b>L (Hr.)</b>	<b>T (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	0	0	3

### **Course Objectives:**

1. To be acquainted with about the important concepts and characteristics of accounting.
2. To study the application of accounting in the general business environment.
3. To prepare the estimate for various business activities such as purchase, sale, production and cash budgets.
4. To provide a comprehensive treatment of accounting principles, technique and practices.
5. To get the students acquainted with fundamental concepts and processes of accounting so that they are able to appreciate the nature of item presented in the annual accounts of an organization.

### **Syllabus**

## **Unit I**

**Accounting Concept:** Basic of Accounting, Meaning and nature of accounting, Scope of financial accounting, Interrelationship of Accounting with other disciplines, Branches of Accounting, Accounting concepts and convention, Indian Accounting Standards

## **Unit II**

**Journal, Ledger and Trial Balance:** Journal: Meaning of Journal, Advantages, and Subdivision. Ledger: Meaning, subdivision, Mechanics of Posting, balancing of Ledger accounts. Trial Balance: Objectives, Defects of trial balance, Errors disclosed by trial balance, preparation and locating errors.

## **Unit III**

**Cash Book and Subsidiary books of Accounting:** Kinds of cashbook, Purchase daybook, Sales daybook, Bills receivable book, Bills payable book.

## **Unit IV**

**Depreciation:** Causes of Depreciation, Meaning of Depreciation Accounting, Method of Recording depreciation, Methods of Providing Depreciation.

## **Unit V**

**The Scope and Method of Economics, the Economic Problem:** Scarcity & Choice, The Price Mechanism, And Demand & Supply Equilibrium: The Concept of Elasticity and its Applications. The Production Process: Output decisions – Revenues Costs and Profit Maximization. Laws of returns & Returns to Scale: Economics and Diseconomies of scale.

## **Course Outcomes (COs):**

**Upon successful completion of this subject students should be able to:**

CO1: Get the Knowledge about the important concepts and characteristics of accounting.

CO2: Able to understand the budget preparation and control of a company.

CO3: Be prepared of fiscal policies of the organization.

CO4: Learn about Inventory Systems, Methods of Valuation of Inventories.

CO5: Learn about Cash Journal, Petty Cash Book, Purchase Journal, Purchase Return, Sales Journal and Sales Return Journal.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

Course Outcome	Program Outcome											
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1		H										
CO2			H						M			
CO3					H							M
CO4	M						H					
CO5		H			M					L		

H = Highly Related; M = Medium; L = Low

### Text Books

1. Maheshwari, S.N. and Maheshwari, S. K., (2009) An Introduction to Accountancy, Eighth Edition, Vikas Publishing House.

### Reference Books

1. Ahuja H.L., "Business Economics", S.Chand & Co., New Delhi, 20012.
2. Financial Accounting: A Managerial Perspective" by Narayanaswamy
3. Tulsian, P.C., (2009) Financial Accountancy, 2nd edition, Pearson Education.
4. Ferfuson P.R., Rothchild, R and Ferguson G.J."Business Economics" Mac-millan, Hampshire, 1993.

**Course Name: Android Application Development**

**Course Code: BCA141A**

<b>L (Hr.)</b>	<b>T (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	0	0	3

**Course Objectives:**

1. To understand the Android application architecture, including the roles of the task stack, activities, and services.
2. To be able to build user interfaces with fragments, views, form widgets, text input, lists, tables, and more.
3. To be able to use advanced UI widgets for scrolling, tabbing, and layout control.
4. To be able to store application data on the mobile device, in internal or external storage locations.
5. To be able to support user-specific preferences using the Android Preferences API.

**Unit I**

**Introduction to Android:** Android Versions, Features of Android, Architecture of Android  
**Obtaining the Required Tools:** Android Studio, Android SDK, Creating Android Virtual Devices (AVDs), Launching Your First Android Application.

**Unit II**

Exploring the IDE, Using Code Completion, Debugging Your Application:- Setting Breakpoints, Navigating Paused Code, Publishing Your Application:- Generating a Signed APK

**Unit III**

**Android User Interface:-** Understanding the Components of a Screen: Views and ViewGroups, LinearLayout (Horizontal) and LinearLayout (Vertical), RelativeLayout, ScrollView, Using Basic Views: TextView, Button, ImageButton, EditText, CheckBox, ToggleButton, RadioButton, and RadioGroup Views, Using List Views to Display Long Lists: ListView, Using the Spinner View.

**Unit IV**

**Activities, Fragments, and Intents:-** Understanding Activities: Applying Styles and Themes to an Activity, Hiding the Activity Title, Displaying a Dialog Window, Displaying a Progress Dialog, Linking Activities Using Intents: Open an Activity, Returning Results from an Intent, Fragments: Life Cycle of a Fragment, Adding Fragments Dynamically, Interactions Between Fragments, Understanding the Intent Object

**Unit V**

Saving and Loading User Preferences: Accessing Preferences Using an Activity, Modifying, Programmatically Retrieving and Modifying the Preferences Values, Creating and Using Databases: Creating the DB Adapter Helper Class, Using the Database Programmatically

**Course Outcomes (COs):**

**Upon successful completion of this subject students should be able to:**

CO1: Student can setup Android Environment, Well understanding of Android Architecture.

CO2: Capable to customize IDE, can insert breakpoint and debugging, learn application deployment.

CO3: Learn to create different views of activity and manage layout for different activity.

CO4: Knowledge of Activity life cycle, and fragment. How intent will work for communication.

CO5: Can store Activity data on local device with database helper class.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

Course Outcome	Program Outcome											
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1		L	H				M		M			
CO2		H			L			M				L
CO3	H				H	M			L			



CO4			H	H			L	M				
CO5		L							H		M	

H = Highly Related; M = Medium; L = Low

**Text Books**

1. Building Android application with HTML, CSS, and JavaScript 2<sup>nd</sup> Edition O'Reilly's Publication
2. Beginning Android Programming with Android Studio

**Reference Books**

1. Android Application Development Cookbook.
2. Android 4.2 App Development Essentials
3. Android Application Development for Dummies

**Course Name: Information Security and Cyber Law**

**Course Code: BCA142A**

<b>L (Hr.)</b>	<b>T (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	1	0	4

**Course Objectives:**

1. To learn today's increasing network security threats and explain the need to implement a comprehensive security policy to mitigate the threats.

2. To provide extended security using authentication, Substitution Techniques, Transposition Techniques, Encryption and Decryption
3. To introduce security services for email and email protocols, Digital Certificates, Private Key Management, E-mail Security and Wireless Application Protocol (WAP) Security
4. To be able to understand awareness of Prohibited actions on Cyber, Cyber Squatting Banking/Credit card related crime E-commerce.
5. To gain the knowledge about Cyber Crime and Prohibited actions on Cyber.

## **Syllabus**

### **Unit I**

**Introduction to the Concepts of Security:** The need for security, Security Approaches, Principles of Security, Types of Attacks. **Cryptographic Techniques:** Plain Text and Cipher Text, Substitution Techniques, Transposition Techniques, Encryption and Decryption, Symmetric and Asymmetric Key Cryptography, Key Range and Key Size, Possible Types of Attacks.

### **Unit II**

**Computer-based Symmetric Key Cryptographic Algorithms:** Algorithm Types and Modes, An overview of Symmetric Key Cryptography, DES, International Data Encryption, Algorithm (IDEA), RC5, Blowfish, AES, Differential and Linear Cryptanalysis.

### **Unit III**

**Computer-based Asymmetric Key Cryptography:** Brief History of Asymmetric Key Cryptography, An overview of Asymmetric Key Cryptography, The RSA Algorithm, Symmetric and Asymmetric Key Cryptography Together, Digital Signatures, Knapsack Algorithm, Some other Algorithms.

### **Unit IV**

**Public Key Infrastructure:** Digital Certificates, Private Key Management, The PKIX Model, Public Key Cryptography Standards, XML, PKI and Security. **Internet Security Protocols:** Basic Concepts, Secure Socket Layer, SHTTP, Time Stamping Protocol E-mail Security, Wireless Application Protocol (WAP) Security.

## Unit V

**Prohibited Actions on Cyber:** Pornography, IPR violations: software piracy, copyright infringement, trademarks violations, theft of computer source code, patent violations, Cyber Squatting Banking/Credit card Related crime E-commerce/ Investment Frauds.

### Course Outcomes (Cos):

**Upon successful completion of this subject students should be able to:**

CO1: Demonstrate the threats in networks and security concepts, Plain Text and Cipher Text, Substitution Techniques, Transposition Techniques, Encryption and Decryption.

CO2: Apply authentication applications in different networks.

CO3: Understand security services for email, the RSA Algorithm, Symmetric and Asymmetric Key Cryptography Together and Digital Signatures.

CO4: Know of Digital Certificates, Private Key Management, E-mail Security and Wireless Application Protocol (WAP) Security

CO5: Awareness of Prohibited actions on Cyber, Cyber Squatting Banking/Credit card related crime E-commerce.

### MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcome	Program Outcome											
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1				H						M		
CO2		H										
CO3							M					M
CO4	M				H						M	
CO5		H						M		L		

H = Highly Related; M = Medium; L = Low

### **Text Books**

1. William Stallings, —Cryptography and Network Security, Prentice Hall, New Delhi, 2006.
2. Charlie Kaufman, Radia Perlman, Mike Speciner, —Network Security: Private Communication in a Public Network, Pearson Education, New Delhi, 2004.

### **Reference Books**

1. Neal Krawetz, —Introduction to Network Security, Thomson Learning, Boston, 2007.
2. Bruce Schneier, —Applied Cryptography, John Wiley & Sons, New York, 2004.
3. Frontiers of of Electronic Commerce Kalakota and Whinston Addison Wesley

**Course Name: Advance Java**

**Course Code: BCA143A**

<b>L (Hr.)</b>	<b>T (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
3	0	0	3

### **Course Objectives:**

1. To be able to exhaustive coverage of advanced topics on Java from tools to enterprise Java
2. To provide ample application-based examples, with step-by-step explanations
3. To provide thorough understanding of each topic through extensive examples along with the program codes and screenshots
4. To provide relevant software installation and configuration information wherever necessary
5. To be able to work with comprises keywords, objective-type questions (with answers) and subjective-type questions for students at the end of all the chapters

### **Syllabus**

#### **Unit I**

**Core Java Recap:** Introduction, Java Evolution and history, Classes and Objects, Functions & Interfaces, Inheritance, keywords: Static, Final, Super, Packages, Applets

## **Unit II**

**Exception Handling, Multi-threading & Garbage Collection:** Limitations of Error handling – Advantages of Exception Handling Types of Errors – Basics of Exception Handling – try blocks – throwing an exception – catching an exception – finally statement, tracing stack, custom exception classes.

Multi-threading : Introduction, Main Thread, Creating Thread, Interrupting Thread, Suspending and Resuming, Thread Priority, Synchronization & Deadlock, Inter-thread Communication Garbage Collection: Introduction, Size, Accessing GC from Java Program

## **Unit III**

**AWT:** AWT Class Hierarchy, Creating Container, Adding Components, Layout, AWT components, Event Handling, Dialog Boxes, Scrollbar, Menu.

## **Unit IV**

**Swings & Input/Output:** Swing Containment Hierarchy, Swing Components, Methods of Important Event Listener Interfaces Streams, Formatting, Data Streams, Object Stream, Reading/writing Arrays via Streams, Pipes, File I/O, Path, File

## **Unit V**

**JDBC:** JDBC Drivers, JDBC Architecture, JDBC Classes and Interfaces, Loading a Driver, Making a Connection, Execute SQL Statement, SQL Statements, Retrieving Result, Getting Database Information, Metadata

## **Course Outcomes (Cos):**

**Upon successful completion of this subject students should be able to:**

CO1: Revise object oriented features of java language and develop java applet programming using various techniques.

CO2: Handling exceptions and develop multi- threaded applications.

CO3: Develop applications using collection framework and concepts of generic programming.

CO4: Develop applications using Abstract Window Toolkit

CO5: Demonstrate the connectivity with SQL through JDBC and describe the JDBC architecture.

**MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:**

Course Outcome	Program Outcome											
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1		H	H						M			
CO2			M			L				M		
CO3				H								M
CO4	L							H				
CO5		M			H					L		

H = Highly Related; M = Medium; L = Low

**Text Books**

1. Sachin Malhotra, Saurabh Chaudhary, "Programming in Java", Second Edition, Oxford University Press, 2014.
2. Advance java programming, Oxford, Uttam Kumar Roy, April 2015.

**Reference Books**

1. Programming with Java A Primer, E. Balaguruswamy Tata McGraw Hill Companies
2. Java Programming Language Ken Arnold Pearson
3. The complete reference JAVA2, Herbert schildt. TMH

**Course Name: Android Application Development Lab**

**Course Code: BCA145A**

<b>L (Hr.)</b>	<b>T (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
0	0	2	2

**Course Objectives:**

The purpose of this course is to enhance the practical knowledge based on prescribed theory course. The students will be able to enhance their analyzing and problem solving skills after implementation of all the given experiments.

**List of Experiment**

Following are the list of sample programs. Students have to perform accordingly.

1. Develop an application that uses GUI components, Font and Colors.
2. Develop an application that uses Layout Managers and event listeners.
3. Develop a native calculator application.
4. Write an application that draws basic graphical primitives on the screen.
5. Develop an application that makes use of database.
6. Develop an application that makes use of RSS Feed.
7. Implement an application that implements Multi threading.
8. Develop a native application that uses GPS location information.
9. Implement an application that writes data to the SD card.
10. Implement an application that creates an alert upon receiving a message.
11. Write a mobile application that creates alarm clock.

**Course Name: Advance Java Lab**

**Course Code: BCA146A**

<b>L (Hr.)</b>	<b>T (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
0	0	2	2

**Course Objectives:**

The purpose of this course is to enhance the practical knowledge based on prescribed theory course. The students will be able to enhance their analyzing and problem solving skills after implementation of all the given experiments.

**List of Experiment**

1. Write a program to sum and average of five marks.
2. Write a program to describe functionality of Arithmetic operations using switch.
3. Write a program to write names of 10 natural numbers.
4. Write a program to sum of 'n' natural numbers.
5. Write a program to find Factorial of a number.
6. Write a program to sum of 'n' numbers.
7. Write a program to sum of two numbers using class and object.
8. Write a program to prepare mark sheet of a student.
9. Write a program to find area of rectangle using constructor.
10. Write a program to find area of room.
11. To write a java program to find the sum of digits of a given number
12. To write a java program to find the first 15 terms of Fibonacci sequence.
13. To write a java program to print the Armstrong numbers.
14. Write a Java program that prints all real solutions to the quadratic equation  $ax^2+bx+c = 0$ .  
Read in a, b, c and use the quadratic formula. If the discriminate  $b^2-4ac$  is negative, display a message stating that there are no real solutions?



15. The Fibonacci sequence is defined by the following rule. The first 2 values in the sequence are 1, 1. Every subsequent value is the sum of the 2 values preceding it. Write a Java program that uses both recursive and non-recursive functions to print the nth value of the Fibonacci sequence?
16. Write a Java program that prompts the user for an integer and then prints out all the prime numbers up to that Integer?
17. Write a Java program that checks whether a given string is a palindrome or not. Ex: MADAM is a palindrome?
18. Write a Java program that reads a line of integers and then displays each integer and the sum of all integers.
19. To write a java program to show how a class implements two interfaces.
20. To write a java program to show that the variables in an interface are implicitly static and final and methods are automatically public.
21. Write a program to implement multilevel inheritance and method overriding.
22. Write a program to illustrate class member access for packages and also implement interfaces.
23. To write a java applet program to change the color of a rectangle using scroll bars to change the value of red, green and blue
24. To write an applet program for creating a simple calculator to perform Addition, subtraction, Multiplication and Division using Button, Label and TextField component.
25. Write a program in Java to describe concepts of handling exceptions.
26. Write a program in Java to describe Exception Handling with all keywords
27. Write a program to describe the concept of Collection framework - an ArrayList.
28. Write a program to describe Java Generics using Map.
29. Write a program in Java in AWT to design login form.
30. Write a Java program to draw a rectangle using swing.
31. Write a Java swing program to print a wave form with output.
32. Write a Enable and Disable button in Java swing.
33. Write a java program that connects to a database using JDBC and does add, deletes, modify and retrieve operations.
34. Write a program to update student information using jdbc connection.
35. Write a program to create an applet of a moving banner.
36. Develop static pages (using only HTML) of an online Book store.

**Course Name: Introduction to PL/SQL Lab**

**Course Code: BCA138A**

<b>L (Hr.)</b>	<b>T/P (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
0	0	2	2

**Course Objectives**

The purpose of this course is to enhance the practical knowledge based on prescribed theory course. The students will be able to enhance their analyzing and problem solving skills after implementation of all the given experiments.

**List of Sample Programs**

1. Create a relational database that contains the following tables and insert 10 records into these tables. Solved the

Following SQL Queries:

student (**sid**, sname, sex, age, year, gpa)

dept (**dname**, numphds)

prof (**pname**, dname)

course (**dname**, **cno**, cname)

major (**dname**, **sid**)

section (**dname**, **cno**, **sectno**, pname)

enroll (**sid**, **dname**, **cno**, **sectno**, grade)

- i. Print the names of professors who work in departments that have fewer than 50 PhD students.
- ii. Print the names of the students with the lowest GPA.
- iii. For each Computer Sciences class, print the class number, section number, and the average gpa of the students enrolled in the class.

- iv. Print the names and section numbers of all classes with more than six students enrolled in them.
  - v. Print the name(s) and sid(s) of the student(s) enrolled in the most classes.
  - vi. Print the names of departments that have one or more majors who are under 18 years old.
  - vii. Print the names and majors of students who are taking one of the College Geometry courses.
  - viii. For those departments that have no major taking a College Geometry course print the department name and the number of PhD students in the department.
  - ix. Print the names of students who are taking both a Computer Sciences course and a Mathematics course.
  - x. Print the age difference between the oldest and the youngest Computer sciences major.
  - xi. For each department that has one or more majors with a GPA under 1.0, print the name of the department and the average GPA of its majors.
  - xii. Print the ids, names and GPAs of the students who are currently taking **all** the Civil Engineering courses.
2. Create the 3 structured record data type named as address\_type(street, city, state, pin\_code), person\_type(firstname, lastname, DOB) and business\_type( title, company). Create 2 tables based on person\_type and business\_type. Create one more table which is referred by first two tables. ( Reference type)
  3. Write a program to enter a number and find the factorial of the number.
  4. Write a code to create a type with an array of number 10.
  5. Write a PL/SQL code to take the input from the user for 3 subjects and calculate total marks and percentage.
  6. Write a program to create a function for add two numbers and call it by passing the values as an argument.
  7. Write a program to enter two number and find greater number using function
  8. Write a program to create a PL/SQL function for count the number of records in any existing table.
  9. Write a PL/SQL code to create a table using the already created array type and insert 5 records.
  10. Write a PL/SQL code to enter two numbers and perform the arithmetic operations. (Addition, Subtraction, Multiplication, Division).

**Course Name: Minor Project**

**Course Code: BCA147A**

<b>L (Hr.)</b>	<b>T (Hr.)</b>	<b>Pr (Hr.)</b>	<b>Credits</b>
0	0	2	2

**Course Objectives:**

The purpose of minor project is to enhance the practical knowledge based on prescribed languages which are introduced in this program. The students will be able to enhance their analyzing and problem solving skills after implementation of all the given experiments.

# List of Courses for Specialization in Data Science & Data Analytics

## OVERVIEW OF DATA SCIENCE AND FOUNDATION OF DATA ANALYTICS

Credit: 2-0-1
Hours: 4 per Week

*Objectives: The objective of this course is to teach students the vital data science concepts and tasks that occupy the data scientist*

### UNIT – I

**Introduction to Data Science:** Defining Data Science and Big Data, Benefits and Uses of Data Science and Big Data, Facets of Data, Structured Data, Unstructured Data, Natural Language, Machine- generated Data, Graph based or Network Data, Audio, Image, Video, Streaming data, Data Science Process, Big data ecosystem and data science, Distributed file systems, Distributed programming framework, data integration framework, machine learning framework, No SQL Databases, scheduling tools, benchmarking tools, system deployments

## **UNIT – II**

**Data Science Processes:** Six steps of data science processes, define research goals, data retrieval, cleansing data, correct errors as early as possible, integrating – combine data from different sources, transforming data, exploratory data analysis, Data modelling, model and variable selection, model execution, model diagnostic and model comparison, presentation and automation.

## **UNIT – III**

**Introduction to Machine Learning:** What is Machine Learning, Learning from Data, History of Machine Learning, Big Data for Machine Learning, Leveraging Machine Learning, Descriptive vs Predictive Analytics, Machine Learning and Statistics, Artificial Intelligence and Machine Learning, Types of Machine Learning – Supervised, Unsupervised, Semi-supervised, Reinforcement Learning, Types of Machine Learning Algorithms, Classification vs Regression Problem, Bayesian, Clustering, Decision Tree, Dimensionality Reduction, Neural Network and Deep Learning, Training machine learning systems

## **UNIT – IV**

**Introduction to AI:** What is AI, Turing test, cognitive modelling approach, law of thoughts, the relational agent approach, the underlying assumptions about intelligence, techniques required to solve AI problems, level of details required to model human intelligence, successfully building an intelligent problem, history of AI

## **UNIT – V**

**Introduction to Data Handling:** Introduction to Data and their uses, Overview of Data analysis, Ribbon|Workbook|Worksheets|Format Cells|Find & Select|Templates|Data Validation|Keyboard Shortcuts|Print|Share|Protect, Working with statistical formulas & functions, Data Validation & data models, Data Analysis using statistical methods

## DATA ANALYSIS USING PYTHON

Credit: 2-0-1
No of Hours : 4 Per Week

*Objectives: The objective of this course is to teach students the concepts of Python Programming Language with Libraries*

### UNIT – I

**Python programming Basic:** Python interpreter, IPython Basics, Tab completion, Introspection, %run command, magic commands, matplotlib integration, python programming, language semantics, scalar types. Control flow

**Data Structure, functions, files:** tuple, list, built-in sequence function, dict, set, functions, namespace, scope, local function, returning multiple values, functions are objects, lambda functions, error and exception handling, file and operation systems

### UNIT – II

**NumPy: Array and vectorized computation:** Multidimensional array object. Creating ndarrays, arithmetic with numpy array, basic indexing and slicing, Boolean indexing, transposing array and swapping axes, universal functions, array-oriented programming with arrays, conditional logic as arrays operations, file input and output with array

### UNIT – III

**Pandas:** Pandas data structure, series, DataFrame, Index Object, Reindexing, dropping entities from an axis, indexing, selection and filtering, integer indexes, arithmetic and data alignment, function application and mapping, sorting and ranking, correlation and covariance, unique values, values controls and membership, reading and writing data in text format

### UNIT -IV

**Visualization with Matplotlib:** Figures and subplots, colors, markers, line style, ticks, labels, legends, annotation and drawing on subplots, matplotlib configuration

## UNIT – V

**Plotting with pandas and seaborn:** line plots, bar plots, histogram, density plots, scatter and point plots, facet grids and categorical data



**PROBABILISTIC MODELLING AND REASONING WITH**  
**PYTHON**

Credit: 2-0-1
No of Hours: 4 per week

*Objectives: The objective of this course is to teach students the concepts of Statistics, probability, probability distribution, and other statistical methods to solve various engineering problems*

**UNIT – I**

**Introduction to Statistics:** Introduction to Statistics. Role of statistics in scientific methods, current applications of statistics.

**Scientific data gathering:** Sampling techniques, scientific studies, observational studies, data management.

**Data description:** Displaying data on a single variable (graphical methods, measure of central tendency, measure of spread), displaying relationship between two or more variables, measure of association between two or more variables.

**UNIT – II**

**Probability Theory:** Sample space and events, probability, axioms of probability, independent events, conditional probability, Bayes' theorem.

**Random Variables:** Discrete and continuous random variables. Probability distribution of discrete random variables, binomial distribution, poisson distribution. Probability distribution of continuous random variables, The uniform distribution, normal (gaussian) distribution

**UNIT -III**

**Point Estimations:** Methods of finding estimators, method of moments, maximum likelihood estimators, bayes estimators. Methods of evaluating estimators, mean squared error, best unbiased estimator, sufficiency and unbiasedness

**Interval Estimations:** Confidence interval of means and proportions, Distribution free confidence interval of percentiles

#### **UNIT - IV**

**Test of Statistical Hypothesis and p-values:** Tests about one mean, tests of equality of two means, test about proportions, p-values, likelihood ratio test, Bayesian tests

#### **UNIT – V**

**Univariate Statistics using Python:** Mean, Mode. Median, Variance, Standard Deviation, Normal Distribution, t-distribution, interval estimation, Hypothesis Testing, Pearson correlation test, ANOVA F-test

# R PROGRAMMING FOR DATA SCIENCE AND DATA

## ANALYSIS

Credit: 2-0-1
Hours: 4 per Week

*Objectives: The objective of this course is to teach students R Programming Language, basic functions in R programming language and critical techniques*

### UNIT – I

**Getting Started with R and R Workspace:** Introducing R, R as a programming Language, the need of R, Installing R, RStudio, RStudio's user interface, console, editor, environment pane, history pane, file pane, plots pane, package pane, help and viewer pane

### UNIT – II

R Workspace, R's working directory, R Project in R Studio, absolute and relative path, Inspecting an Environment, Inspect existing Symbols, View the structure of object, Removing symbols, Modifying Global Options, Modifying warning level, Library of Packages, Getting to know a package, Installing a Package from CRAN, Updating Package from CRAN, Installing package from online repository, Package Function, Masking and name conflicts

### UNIT – III

**Basic Objects and Basic Expressions:** Vectors, Numeric Vectors, Logical Vectors, Character Vectors, subset vectors, Named Vectors, extracting element, converting vector, Arithmetic operators, create Matrix, Naming row and columns, subsetting matrix, matrix operators, creating and subsetting an Array, Creating a List, extracting element from list, subsetting a list, setting value, creating a value of data frame, subsetting a data frame, setting values, factors, useful functions of a data frame, loading and writing data on disk, creating a function, calling a function, dynamic typing, generalizing a function. Assignment Operators, Conditional Expression, using if as expression and statement, using if with vectors, vectorized if: ifelse, using switch, using for loop, nested for loop, while loop

## **UNIT – IV**

**Working with Basic Objects and Strings:** Working with object function, getting data dimensions, reshaping data structures, iterating over one dimension, logical operators, logical functions, dealing with missing values, logical coercion, math function, number rounding functions, trigonometric functions, hyperbolic functions, extreme functions, finding roots, derivatives and integration, Statistical function, sampling from a vector, Working with random distributions, computing summary statistics, covariance and correlation matrix, printing string, concatenating string, transforming text, Formatting text, formatting date and time, formatting date and time to string, finding string pattern, using group to extract data, reading data

## **UNIT – V**

**Working with Data – Visualize and Analyze Data:** Reading and Writing Data, importing data using built-in-function, READR package, export a data frame to file, reading and writing Excel worksheets, reading and writing native data files, loading built-in data sets, create scatter plot, bar chart, pie chart, histogram and density plots, box plot, fitting linear model and regression tree

**FOUNDATION OF MACHINE LEARNING AND PATTERN  
RECOGNITION**

Credit: 2-0-1
No of Hours: 4 per week

*Objectives: The objective of this course is to teach students the basic concepts of machine learning, supervised learning, unsupervised learning, and reinforcement learning*

**UNIT – I**

**Introduction:** Learning systems, real world applications of machine learning, why machine learning, variable types and terminology, function approximation

**Types of machine learning:** Supervised learning, unsupervised learning, reinforcement learning

**Important concepts of machine learning:** Parametric vs non-parametric models, the trade-off between prediction accuracy and model interpretability, the curse of dimensionality, measuring the quality of fit, bias-variance trade off, overfitting, model selection, no free lunch theorem

**UNIT – II**

**Linear Regression:** Linear regression, estimating the coefficients, accessing the accuracy of coefficient estimates, accessing the accuracy of the model

**Classification:** Logistic regression, estimating regression coefficients, making predictions, multiple logistic regressions, linear discriminant analysis, bayes' theorem of classification,

**UNIT – III**

**Resampling Methods, Model Selection and Regularization:** Cross-validation, leave-one-out cross-validation, k-fold cross-validation, the bootstrap, subset selection, shrinkage methods, ridge and lasso regression, dimension reduction methods, principal components regression

**Tree Based Methods:** Advantages and disadvantages of trees, regression Trees, classification trees, bagging, random forest, boosting

#### **UNIT – IV**

**Support Vector Machine:** Maximum margin classifier, classification using a separating hyperplane, the maximal margin classifier, support vector classifier, support vector machines, classification with non-linear decision boundaries, support vector machine

#### **UNIT – V**

**Unsupervised Learning:** Principle component analysis, what are principal components, clustering methods, k-means clustering, hierarchical clustering,

## DATA VISUALIZATION USING TABLEAU

Credit: 2-0-1
No of Hours : 4 Per Week

### UNIT - I

**INTRODUCTION TO DATA HANDLING** Overview of Data analysis, Introduction to Data visualization, Working with statistical formulas - Logical and financial functions , Data Validation & data models, Power Map for visualize data , Power BI-Business Intelligence , Data Analysis using statistical methods, Dashboard designing.

### UNIT - II

**INTRODUCTION TO DATA MANIPULATION USING FUNCTION:** Heat Map, Tree Map, Smart Chart, Azure Machine learning , Column Chart, Line Chart , Pie,Bar, Area, Scatter Chart, Data Series, Axes , Chart Sheet , Trendline , Error Bars, Sparklines, Combination Chart, Gauge, Thermometer Chart , Gantt Chart , Pareto Chart etc ,

### UNIT – III

Frequency Distribution, Pivot Chart, Slicers , Tables: Structured References, Table Styles , What-If Analysis: Data Tables, Goal Seek, Quadratic Equation , Transportation Problem, Maximum Flow Problem, Sensitivity Analysis, Histogram, Descriptive, Statistics, Anova, F-Test, t-Test, Moving, Average, Exponential Smoothing | Correlation model | Regression model, Practical Lab

### UNIT – IV

**TABLEAU SOFTWARE: GETTING STARTED WITH TABLEAU SOFTWARE:** What is Tableau? What does the Tableau product suite comprise of? How Does Tableau Work? Tableau Architecture, What is My Tableau Repository? Connecting to Data & Introduction to data source concepts, Understanding the Tableau workspace, Dimensions and Measures, Data Types & Default Properties, Building basic views, Saving and Sharing your work-overview, Practical Lab

## **UNIT - V**

**TABLEAU BUILDING VIEWS (REPORTS):** Date Aggregations and Date parts, Cross tab & Tabular charts, Totals & Subtotals, Bar Charts & Stacked Bars, Trend lines, Forecasting, Filters, Context filters, Line Graphs with Date & Without Date, Tree maps, Scatter Plots



## **BIG DATA ANALYTICS WITH HADOOP**

Credit: 2-0-1
No of Hours: 4 per Week

*Objectives: The objective of this course is to teach students the conceptual framework of Big Data, Virtualization, MapReduce, HDFS, Pig, Hive, Spark, ZooKeeper, HBase*

### **UNIT – I**

**Big Data:** Fundamentals of Big Data, defining big data, building successful big data management architecture, big data journey

**Big Data Types:** Structured and unstructured data types, real time and non-real time requirements

**Distributed Computing:** History of distributed computing, basics of distributed computing

### **UNIT – II**

**Big Data Technology Foundation:** Big Data stack, redundant physical infrastructure, security infrastructure, operational databases, organising data services and tools, analytical data warehouse, big data analytics

**Virtualization:** Basics of virtualization, hypervisor, abstraction and virtualization, implementing virtualization with big data

**Cloud and Big Data:** Defining cloud, cloud deployment and delivery models, cloud as an imperative for big data, use the cloud for big data

### **UNIT – III**

**Operational Databases:** Relational database, nonrelational database, key-value pair databases, document databases, columnar databases, graph databases, spatial databases

**MapReduce Fundamentals:** Origin of MapReduce, map function, reduce function, putting map and reduce together, optimizing map reduce

**Hadoop:** Discovering Hadoop, Hadoop distributed file system, Hadoop MapReduce, Hadoop file system, dataflow, Hadoop I/O, data integrity, compression, serialization, file-based data structure

#### **UNIT – IV**

**Avro:** Avro data types and schemas, in-memory serialization and deserialization, avro datafiles, schema resolution

**Pig:** Comparison with databases, pig latin, user defined functions, data processing operators

**Hive:** Running hive, comparison with traditional databases, HiveQL, tables, querying data, user- defined functions

#### **UNIT – V**

**Spark:** Resilient distributed datasets, shared variables, anatomy of a spark job run, executors and cluster managers,

**HBase:** HBasics, concepts, clients, HBase vs RDBMS, Praxis

**ZooKeeper:** ZooKeeper services, building application with ZooKeeper

## SCALA FOR DATA SCIENCE

Credit: 2-0-1
Hours: 4 per Week

*Objectives: The objective of this course is to teach students Scala Programming Language, basic functions in Scala programming language and critical techniques*

### UNIT – I

**Scala Language:** Getting to know Scala programming language, Scala and Java, Statically typed language, Apache Spark and Scala, Scala Performance Benefits, Installing Scala, Using Scala REPL/Shell, getting help from Scala shell, Hello World, Paste mode, retrieving history, auto-complete feature, exiting from Scala REPL

### UNIT – II

**Variables, Data Types, Conditional Statements:** Immutability of variables, define mutable and immutable variables, mutability and type safety, Specifying types for variables, Scala Identifier rules, naming conventions, Scala data types, Boolean types, string type, multiline strings, string operations, string concatenation, string interpolation, length of string, splitting string, extracting part of string, index of character of strings, the ANY type, type casting, Boolean expressions, conditional statement in Scala, nested IF/ELSE statement, pattern matching,

### UNIT – III

**Code Blocks, Functions, Collections:** Code Blocks in Scala, Why use functions in Scala, understanding functions in Scala, define and invoke a function, functions with multiple parameters, positional parameters, functions with no argument, single-line function, passing function as argument, anonymous function

## UNIT – IV

Collections in Scala, Understanding List, list size, convert list to string, iterating over list, map function and collection, foreach, reduce operation, list equality, create set, indexing map, manipulating maps, understanding tuples, indexing tuples, mutable collections, nested collections

## UNIT – V

**Loops, Packages, Classes and Exceptional Handling:** For loop, While loop, Breaking Loop iteration, classes and objects in Scala, Create classes and objects, singleton objects, case classes, equality checks, classes and packages, avoid name space collusion, importing package, fundamental of exception handling, type inferences and exception handling, try, catch, finally, Scala built tool (SBT), Compile Scala applications,