1

CHOICE BASED CREDIT SYSTEM
Semester Scheme with Multiple Entry and Exit Options for
Four Year Undergraduate Programmed

COURSE STRUCTURE
&
DETAILED SYLLABUS
For
BACHELOR OF COMPUTER APPLICATIONS

MANIPUR UNIVERSITY
CANCHIPUR, IMPHAL

Course Structure (Draft)
Model A

Bachelor of Computer Application is a 4(four) year undergraduate program aimed to shape students in contemporary knowledge of fundamentals to comprehend applicative technology towards efficient solutions for industrial and real-life problems. The course curriculum inculcates social awareness, communication skills & professionalism to work as a team in maintaining diverse environments towards the betterment of society. There is a growing need for qualified computer engineers and a BCA can help you create a multi-faceted career in the industry. BCA degree can be employed in these sectors: healthcare, IT, finance, trading, transportation, software, and education.

Course Structure for the 4-Year Undergraduate Program (Bachelor of Computer Applications)						
Semester	Core (Credit)	DSE (Credit)	GEC (Credit)	AECC (Credit)	SEC (Credit)	VAC (Credit)	Semester Credit
I	BCA101(6)			AECC-1 (4)	BCASE-123A/B/C-(4)	VAC-101(2)	24
	BCA102(6)					VAC-	
II	BCA203(6)			AECC-2 (4) Environme	BCASE-224A/B/C (4)	VAC-203(2)	24
	BCA204(6)					VAC-204(2)	
Exit Option with Bachelor's Certificate of Computer Application on Completion of courses equal to a minimum of 46 Credits							
III	BCA305(6)		BCA101 (6)			VAC-305(2)	26
	BCA306(6)						
	BCA307(6)						
IV	BCA408(6)		BCA102 (6)			VAC-406(2)	26
	BCA409(6)						
	BCA410(6)						
Exit Option with Bachelor's Diploma in Computer Application on Completion of courses equal to a minimum of 96 Credits							
V	BCA511(6)	BCA519A/B/C(6) 1(6)	BCA203(6) 3(6)			VAC-507(2)	26
	BCA512(6)						
VI	BCA613(6)	BCA620A/B/C(6)	BCA512(6)			VAC-608(2)	26
	BCA614(6)						
Exit Option with Bachelor's Degree in Computer Application on Completion of courses equal to a minimum of 140 Credits							
VII	BCA715(6)	BCA721A/B/C(6)	BCA410 (6)				24
	BCA716(6)						
VIII	BCA817(6)	BCA822A/B/C(6)	BCA825 (6)				24
	BCA818(6)						
Award of Bachelor's Degree with Honours in Computer Application on Completion of courses equal to a minimum of 182 Credits							

Bachelor of Major & Honours in Computer Applications. Course Structure (Draft)

Semester-Wise Distribution of Courses

A. Discipline Specific Core (DSC) Courses

All the courses have 6 credits with 4 credits of theory (4 hours per week) and 2 credits of practical (4 hours Or 5 (FIVE) credits of theory and 1 (ONE) credit of tutorial

Sl. No.	Paper Code	Semester	Course Name
1	BCA101	I	Fundamentals of Information Technology
2	BCA102	I	Programming using C
3	BCA203	II	Programming using Python
4	BCA204	II	Computer System Architecture
5	BCA305	III	Data Structure using C
6	BCA306	III	Operating System
7	BCA307	III	Discrete Mathematics
8	BCA408	IV	Probability and Statistics
9	BCA409	IV	DBMS
10	BCA410	IV	Web Technologies
11	BCA511	V	Computer Networks
12	BCA512	V	Java Programming
13	BCA613	VI	Theory of Computation
14	BCA614	VI	Financial Accounting
15	BCA715	VII	Design and analysis of Algorithm
16	BCA716	VII	Computer Graphics
17	BCA817	VIII	Artificial Intelligence
18	BCA818	VIII	Software Engineering

B. Discipline Specific Elective Papers: (Credit: 06 each) DSE-1,DSE-2, DSE-3, DSE-4.

All the courses have 6 credits with 4 credits of theory and 2 credits of practical or 5 credits of theory and 1 credit of tutorials.

DSE-1   (Choose any one)

BCA519A	.NET
BCA519B	Network Security
BCA519C	Computer Oriented Numerical Methods

DSE-2 (Choose any one)

BCA620A	Internet of Things (IOT) and its applications
BCA620B	Cloud Computing
BCA620C	Machine Learning

DSE-3 (Choose any one)

BCA721A	Deep Learning
BCA721B	Data Analysis and Visualization
BCA721C	Fundamentals of Data Science

DSE-4 (Choose any one)

BCA822A	Digital Image Processing
BCA822B	Data Mining
BCA822C	Project Work/Dissertation

C. Other Discipline: (Credit: 06 each) - GEC-1 to GEC-6
Skill Enhancement Courses (Credit: 04 each): SEC-1, SEC-2

SEC-1: (Choose any one)

BCASE123A	Mathematical Foundation
BCASE123B	Office Automation Tools
BCASE123C	PC Hardware and maintenance

SEC-2: (Choose any one)

BCASE224A	Linux Environment
BCASE224B	Object oriented programming using C++
BCASE224C	Digital Electronics

D. Ability Enhancement Compulsory Courses
All the courses have 4 credits including Theory/Practical's/Projects

Sl. No.	AECC Paper Code	Semester	AECC Name
1	AECC-101	I	English/MIL
2	AECC-202	II	Environmental Science

E. Value added Courses

Sl. No.	VAC Paper Code	Semester	VAC Name
1	VAC-101	I	Yoga
2	VAC-102	I	Sports
3	VAC-203	II	Culture
4	VAC-204	II	Health Care
5	VAC-305	III	NCC
6	VAC-406	IV	Ethics
7	VAC-507	V	NSS
8	VAC-608	VI	History of Science

F. General Elective Courses
All the Courses have 6 credits with 4 credits of theory and 2 credits of practical. These courses are meant for students of other departmental/disciplines or 5 credits of theory and 1 credit of tutorials.

Sl. No	Paper Code	Semester	GEC Name
1	BCA101	I	Fundamentals of Information Technology
2	BCA102	I	Programming using C
3	BCA203	II	Programming with Python
4	BCA512	V	Java Programming
5	BCA410	VII	Web Technologies
6	BCA825	VIII	Introduction to Multimedia

Contents of Courses for Bachelor degree in Computer Applications
Model B

Semester	Course code	Theory/ Practical	Credit	Paper Titles	Marks		Remarks
					S.A	L.A	
I	BCA101	Theory	4	Fundamentals of information Technology	75	25	
	BCA101 P	Practical	2	-do-	35	15	
	BCA-102	Theory	4	Programming using C	75	25	
	BCA102 P	Practical	2	-do-	35	15	
	BCASE-123A/B/C	Theory and Practical	4	Mathematical Foundation/ Office Automation Tools/ PC Hardware and maintenance	75	25	
II	BCA203	Theory	4	Programming with Python	75	25	
	BCA203 P	Practical	2	-do-	35	15	
	BCA204	Theory	5	Computer System Architecture	75	25	
	BCA204	Tutorial	1				
	BCA204 P	Practical	2	-do-	35	15	
	BCASE-224-A/B/C	Theory and Practical	4	Linux Environment/ Object oriented programming using C++/ Digital Electronics	75 35	25 15	
				Exit Option with Certificate			
III	BCA305	Theory	4	Data Structure using C	75	25	
	BCA305 P	Practical	2	-do-	35	15	
	BCA306	Theory	5	Operating System	75	25	
	BCA306 P	Tutorial	1	-do-			
	BCA307	Theory	5	Discrete Mathematics	75	25	
	BCA307 T	Tutorial	1	-do-			
	BCA101(GEC)	Theory	4	Fundamentals of information Technology	75	25	
	BCA101(GEC)	Practical	2	-do-	35	15	
IV	BCA408	Theory	5	Probability and Statistics	75	25	
	BCA408	Tutorial	1	-do-			
	BCA409	Theory	4	DBMS	75	25	
	BCA409	Practical	2	-do-	35	15	
	BCA410	Theory	4	Web technologies	75	25	
	BCA410	Practical	2	-do-	35	15	

	BCA102(GEC)	Theory	5	Programming using C	75	25	
	BCA102(GEC)	Practical	1	-do-			
Exit Option with Diploma							
V	BCA511	Theory	5	Computer Networks	75	25	
	BCA511	Tutorial	1	-do-			
	BCA512	Theory	4	Java Programing	75	25	
	BCA512	Practical	2	-do-	35	15	
	BCA519A/B/C	Theory	4/5	.NET/ Network Security/Computer Oriented Numerical Methods/	75	25	
	BCA519A/B/C	Practical/ Tutorial	2/1	. NET/ Network Security/Computer Oriented Numerical Methods/	35	15	
	BCA203(GEC)	Theory	5	Programming with Python	75	25	
	BCA203(GEC)	Practical	1	-do-			
VI	BCA613	Theory	5	Theory of Computation	75	25	
	BCA613	Tutorial	1	-do-			
	BCA614	Theory	5	Financial Accounting	75	25	
	BCA614	Tutorial	1	Financial Accountancy			
	BCA620A/B/C	Theory	4/5	Internet of Things (IOT) and its applications/ Cloud Computing/Machine learning	75	25	
	BCA620A/B/C	Practical/ Tutorial	2/1	-do-	35	15	
	BCA512(GEC)	Theory	5	Java Programming	75	25	
	BCA512(GEC)	Practical	1	-do-			
Exit option with Bachelor of Computer Applications							
	BCA715	Theory	4	Design and Analysis of Algorithm	75	25	
	BCA715	Practical	2	-do-	35	15	
	BCA716	Theory		Computer Graphics	75	25	
	BCA716	Practical		-do-	35	15	

VII	BCA721A/B/	Theory	4/5	Deep Learning/ Data Analysis and Visualization/ Fundamentals of Data Science	75	25	
	BCA721A/B/	Practical/ Tutorial	2/1	-do-	35	15	
	BCA410(GEC)	Theory	4	Web Technologies	75	25	
	BCA410(GEC)	Practical	2	-do-	35	15	
VIII	BCA817	Theory	4	Artificial Intelligence	75	25	
	BCA817	Practical	2	-do-	35	15	
	BCA818	Theory	5	Software Engineering	75	25	
	BCA818	Tutorial	1	-do-			
	BCA822A/B/C	Theory	4/5	Digital Image Processing/ Data Mining/ Project Work or Dissertation	75	25	
	BCA822A	Practical/ Tutorial	2/1	-do-	35	15	
	BCA825(GEC)	Theory	4	Introduction to Multimedia	75	25	
	BCA825(GEC)	Practical	2	-do-	35	15	
Award of Bachelor of Honours in Computer Applications							

Courses/Program for Bachelors of Computer Applications

BCA101: Fundamentals of Information Technology (Discipline Specific Core Course)
Credit: 06
Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)
Workload: 4 Lectures (Per Week), 4 Practical (Per Week)

Course Objective
This course is designed to help the students to learn the basic study of computer hardware, operating systems, networking, Internet, databases, etc.

Course Learning Outcomes
On successful completion of the course, students will be able to:
1. Understand the concept of hardware and software.
2. Acquainting with input and output devices.
3. Understand networking concepts and models.
4. Learn and aware of Internet activities.

Detailed Syllabus

UNIT-1: 15 mark
What is Computer? Representation of data/information. What is Data Processing? Characteristics of a Computer System, Evolution of Computer, Generation of computers, Block Diagram of Digital Computer, Classification of Digital Computers Classification of Computers, Functional Components of a computer- Central Processing Unit, Memory-Primary Memory-RAM, ROM, Booting, Secondary Storages Devices : Floppy and Hard Disks, Optical Disks CD-ROM, DVD, Mass Storage Devices : USB thumb drive. Input /Output Devices- Keyboard, Mouse, Trackball, Joystick, Digitizing Tablet, Scanners, Digital Camera, MICR, OCR, OMR, Bar-code Reader, Voice Recognition, Light Pen, Touch Screen, Monitors, Printers & types – Daisy wheel, Dot Matrix, Inkjet, Laser, Line Printer, Plotter, Sound Card and Speakers.

UNIT-II: 15 mark
SOFTWARE : Types of Software, Classification of System and Application Software, System Software - Operating System, Devices Drivers, Application development Packages, Word Processors, Electronic Spreadsheet, Data Base Management System, Presentation packages, Desk Top Publishing Software(DTP), Overview of languages- Machine language, Assembly language, high level languages, Types of high level languages, Generation of languages, Commands , Utility Program-Disk Cleanup Utility, Desk backup Utility, Antivirus Utilities, Customized Application Software.

UNIT-III: 15 mark
NUMBER SYSTEM: Binary, Octal, Decimal, Hexadecimal and Conversion between Number Systems, ASCII Codes, Unicode.

UNIT-IV: 15 mark
MICROPROCESSOR: Basic concepts, Clock speed (MHz, GHz), 16 bit, 32 bit, 64 bit processors: Types: CISC, RISC, Concepts of System Buses, Address Bus, Data Bus, Concepts of Accumulator, Instruction Register, Program counter, Commonly used CPUs and CPU related Terminologies: Intel Pentium Series, Intel Celeron, Cyrix, AMD Series, Xeon, Intel Mobile, Mac Series, CPU Cache, Concept of heat sink and CPU fan, Motherboard; Single, Dual and Multiple Processors.

UNIT-V: 15 mark

NETWORKING & INTERNET: What is network? Need for networking, Evolution of networking, Types of networks, Data Communication & Terminologies, Network Topologies, Network Devices-Modem, Ethernet Card, Hub, Switch, Repeater, Bridge, Router, Gateway; Switching Techniques, Transmission Media, Communication Protocols, Security Concepts. Basic concept of Internet, World Wide Web, Web Browser, Web Server, Web Sites , Web Pages, URL, Domain Names, Hyper Text Mark Up Languages, Internet Address, Electronic Mail, Internet Service Provider, Search Engines.

Practical:

Part A: Hardware

1. Identification of the peripherals of a computer, components in a CPU and their functions.
2. Assembling and disassembling the system hardware components of personal computer.
3. Basic Computer Hardware Trouble shooting.
4. LAN and WiFi Basics.
5. Operating System Installation – Windows OS, UNIX/LINUX, Dual Booting.
6. Installation and Uninstallation of Software – Office Tools, Utility Software (like Anti-Virus, System Maintenance tools); Application Software - Like Photo/Image Editors, Audio Recorders/Editors, Video Editors ...); Freeware, Shareware, Payware and Trialware; Internet Browsers, Programming IDEs,
7. System Configuration – BIOS Settings, Registry Editor, MS Config, Task Manager, System Maintenance, Third-party System Maintenance Tools

Part B: Software

1. Activities using Word Processor Software
2. Activities using Spreadsheets Software.
3. Activities using Presentation Software
4. Activities involving Multimedia Editing (Images, Video, Audio ...)
5. Tasks involving Internet Browsing
6. Flow charts: Installation and using of flowgarithms software for different arithmetic tasks like sum, average, product, difference, quotient and remainder of given numbers, calculate area of Shapes (Square, Rectangle, Circle and Triangle), arrays and recursion.

Text Books:

1. Basics of Information Technology-Sumita Arora,Dhanpat Rai & Co (Pvt.) Ltd.,New Delhi.
2. Foundation of Information Technology – D.S. Yadav, New Age International Publisher.
3. V. Raja Raman, "Introduction to Computers", PHI, 1998.
4. Alex Leon & Mathews Leon, "Introduction to Computers", Vikas Publishing House, 1999. Norton Peter, "Introduction to computers", 4th Ed., TMH, 2001.

References:

1. Computer Applications in Management: Ritendra Goel, D.N. Kakkar, New Age International Publishers Ltd.
2. Computer Technology and Programming: H.K. Gundurao, N.S. Manjunath, M.N. Nachappa, Himalaya Publishing House.
3. Computer Applications in Management: Sanjay Saxena, Prabhapreet Chopra, Vikas Publishing House Pvt. Ltd.
4. Introduction to Computer Science: ITL Education Solutions Ltd. Pearson Education

5. S.K.Basandra, "Computers Today ", Galgotia Publications. Alexis Leon & Mathews Leon, "Fundamentals of Information technology ", Vikas Publishing House, New Delhi.

BCA102: Programming using C (Discipline Specific Core Course)
Credit: 06
Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)
Workload: 4 Lectures (Per Week), 4 Practical (Per Week)

Course Objective

This course is designed to impart knowledge of programming using C language thereby enabling the students to induce thinking and develop logics which will help them to create programs, applications in C. Learning the fundamentals of C programming constructs shall help the students easily switch over to any other languages in future.

Course Learning Outcomes

On successful completion of the course, students will be able to:
1. Understand the art of programming.
2. Solve programming problems.
4. Develop applications
3. Handle external files as well as exceptions.

Detailed Syllabus

UNIT-I: 15 mark
INTRODUCTION TO C PROGRAMMING: The Basic Model of Computation, Algorithms, Flow-charts, Programming Languages, Compilation, Linking and Loading, Testing and Debugging, Documentation. Programming using C: Concept of variables, program statements and function calls from the library, data types, int, char, float etc., declarations and expressions, arithmetic operation, relational and logical operations, C assignment statements, extension of assignment of the operations. C primitive input output functions, C Statements,

UNIT-II: 15 mark
CONDITIONAL STATEMENTS AND LOOPS: Decision making within a program, Conditions, Relational Operators, Logical Connectives, if statement, if-else statement, Loops: while loop, do while, for loop, Nested loops, Infinite loops, Switch statement, structured Programming .

UNIT-III: 15 mark
FUNCTIONS: Top-down approach of problem solving, Modular programming and functions, Standard Library of C functions, Prototype of a function: Formal parameter list, Return Type, Function call, Block structure, Passing arguments to a Function: call by reference, call by value, Recursive Functions, arrays as function arguments. Storage Classes: Scope and extent, Storage Classes in a single source file: auto, extern and static, register, Storage Classes in a multiple source files: extern and static

UNIT-IV: 15 mark
ARRAYS: One dimensional arrays: Array manipulation; Searching, Insertion, Deletion of an element from an array; Finding the largest/smallest element in an array; Two dimensional arrays, Addition/Multiplication of two matrices, Transpose of a square matrix; Null terminated strings as array of characters, Standard library string functions Structures and Unions ,

Structure variables, initialization, structure assignment, nested structure, structures and functions, structures and arrays: arrays of structures, structures containing arrays, unions.

UNIT-V: 15 mark
POINTERS-Address operators, pointer type declaration, pointer assignment, pointer initialization, pointer arithmetic, functions and pointers, Arrays and Pointers, pointer arrays, pointers and structures, dynamic memory allocation. File Processing: Concept of Files, File opening in various modes and closing of a file, Reading from a file, writing onto a file.

Text Books:
1. Balagurusamy, "Programming with ANSI-C", Fourth Edition, 2008, Tata McGraw Hill.
2. Venugopal K. R and Prasad S. R, "Mastering 'C'", Third Edition, 2008, Tata McGraw Hill.
3. B.W. Kernighan & D. M. Ritchie, "The C Programming Language", Second Edition, 2001, Pearson Education

References:
1. Byron S Gottfried "Programming with C" Second edition, Tata McGrawhill, 2007 (Paper back)
2. R.G. Dromey, "How to solve it by Computer", Pearson Education, 2008.
3. Yashwant Kanetkar,, "Let us C", BPB Publications, 2007.
4. Hanly J R & Koffman E.B, "Problem Solving and Programm design in C", Pearson Education, 2009.

Practical for C Programming
1. To find the sum of square of natural number with 20
2. To print 50 odd number starting from 1.
3. To find factorial of a number.
4. To determine whether a number is prime or not.
5. To find the largest of three numbers.
6. To check whether the character entered is an alphabet or a number.
7. To generate first ten elements of Fibonacci series.
8. To find the roots of a quadratic equation.
9. To print perimeter and area of a circle.
10. To calculate the net salary of an employee based on basic pay, DA, TA and PF.
11. To create a menu driven arithmetic calculator to perform basic arithmetic operations on the basis of user operator choice.
12. To check whether the entered character is a vowel or a consonant.
13. To accept 10 numbers from user and print largest number entered.
14. To print a Fahrenheit-to-Celsius conversion and vice versa
15. To evaluate the power series $e^x = 1 + x + x^2/2! + \ldots + x^n/n!$, $0<x<1$
16. To print the following
   1
   2 2
   3 3 3
   4 4 4 4

17. Write a program to display ASCII value of a character.
18. Write a program to check whether a number is perfect or not.
19. Write a program to find out the biggest of three numbers using nested if.

20. A company insures its drivers if either of the following conditions are satisfied • Driver is married. • Driver is an unmarried, male and above 30 years of age. • Driver is unmarried, female and above 25 years of age. Write a program to decide if a driver is to be insured using logical operators.

21. Write a program to read a list of positive integers terminated by -1 and display the odd and even numbers separately and also their respective counts.

22. Write a program to read values of n and x and print the value of y using switch case where a. y=n+x when n=1 b. y=1+x/n when n=2 c. y= n+3x when n=3 d. y=1+nx

23. Write a program to n values of sales and then calculate the commission on sales amount where the commission is calculated as follows: a. If sales <= Rs.500, commission is 5%. b. If sales > 500 but <= 2000, commission is Rs 35 plus 10% above Rs 500. c. If sales > 2000 but <= 5000, commission is Rs 185 plus 12% above Rs.2000. d. If sales > 5000, commission is 12.5%

24. Write a program to find out minimum, maximum, sum and average of n numbers without using array.

25. Program to find mean and standard deviation (SD) for a set of n numbers without using array.

26. Write a program to print the digits of a number in words. (eg. if a number 841 is entered through the keyboard your program should print "Eight Four One".)

27. Write a program to print the PASCAL Triangle up to the n-th row where n is an input to the program.

28. Write a function to return the HCF of two positive integers. Write a main function to read two positive integers and print their HCF and LCM by using the above function.

29. Write a program to convert a decimal number into binary number using function.

30. Write a program to display the result of sine series using function.

31. Write a program to find the sum of the following series $1+x-x^3/3!+x^5/5!-x^7/7!+$. Corrected up to the 3 decimal place.

32. Write a program to read n numbers in a sorted array and insert a given element in a particular position

33. Write functions to compute the factorial of a number using both recursive and non-recursive procedure.

34. Write a program to Multiply two matrices using function

35. Write a program to display the upper Triangle and lower Triangle of a given square matrix using function.

36. Write a function to check if a given square matrix is symmetric or not. Write a main function to implement it.

37. Write a program to read a m x n matrix and calculate the Row sum and Column Sum of the matrix

38. Write a function to read in an integer and print the representation of the number using the sign and magnitude representation scheme using 8 bits. The program should check for overflow/under flow conditions. The left most bit is to be used as the sign bit.

39. Write a program to merge two sorted arrays.

40. Write a program to implement selection sort using function.

41. Write a program to count the number of vowels in a string.

42. Write a program to concatenate two strings using function (without using library function).

43. Write a program to convert a string from upper case to lower case and vice versa.

44. Write a program to swap two numbers using function (pass the pointers)

45. Write a program to sort n number of strings in ascending order using pointer.

46. Write a program using pointers to copy a string to another string variable (without using library function).

47. Declare a structure of a student with details like roll number, student name and total marks. Using this, declare an array with 50 elements. Write a program to read details of n students and print the list of students who have scored 75 marks and above.
48. Write a program to read a text file and count the number of vowels in the text file.
49. Write a program to copy a text file to another file.

<div align="center">

BCA203: Programming Using Python (Discipline Specific Core Course)
Credit: 06
Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)
Workload: 4 Lectures (Per Week), 4 Practical (Per Week)

</div>

Course Objective
This course is designed to introduce the student to the basics of programming using Python. The course covers the topics essential for developing well documented modular programs using different instructions and built-in data structures available in Python.

Course Learning Outcomes
On successful completion of the course, students will be able to:
1. Develop, document, and debug modular python programs to solve computational problems.
2. Select a suitable programming construct and data structure for a situation.
3. Use built-in strings, lists, sets, tuples and dictionary in applications.
4. Define classes and use them in applications.
5. Use files for I/O operations.

Detailed Syllabus

UNIT-I: 10 mark
Introduction to Programming using Python: Structure of a Python Program, Functions, Interpreter shell, Indentation. Identifiers and keywords, Literals, Strings, Basic operators (Arithmetic operator, Relational operator, Logical or Boolean operator, Assignment Operator, Bit wise operator).

UNIT-II: 15 mark
Building blocks of Python: Standard libraries in Python, notion of class, object and method.

UNIT-III: 15 mark
Creating Python Programs: Input and Output Statements, Control statements:-branching, looping, Exit function, break, continue and pass, mutable and immutable structures. Testing and debugging a program

UNIT-IV: 10 mark
Built-in data structures: Strings, lists, Sets, Tuples and Dictionary and associated operations. Basic searching and sorting methods using iteration and recursion.

UNIT-V: 15 mark
Visualization using 2D and 3D graphics: Visualization using graphical objects like Point, Line, Histogram, Sine and Cosine Curve, 3D objects

UNIT-VI: 10 mark
Exception Handling and File Handling: Reading and writing text and structured files, Errors and Exceptions.

14

Text Books:
1. Downey, A.B., (2015), *Think Python–How to think like a Computer Scientist*, 3rd edition.O'Reilly Media.
2. Taneja, S. & Kumar, N., (2017), *Python Programming- A Modular Approach*. Pearson Education.

References:
1. Brown, M. C. (2001). The Complete Reference: Python, McGraw Hill Education.
2. Dromey, R. G. (2006), How to Solve it by Computer, Pearson Education.
3. Guttag, J.V. (2016), Introduction to computation and programming using Python. MIT Press.
4. Liang, Y.D. (2013), Introduction to programming using Python. Pearson Education

Practical
1. Execution of expressions involving arithmetic, relational, logical, and bitwise operators in the shell window of Python IDLE.
2. Write a Python function to produce the outputs such as:

a)
```
  *
* * *
* * * * *
* * *
  *
```
(b)
```
        1
      232
    34543
   4567654
  567898765
```
3. Write a Python program to illustrate the various functions of the "Math" module.
4. Write a function that takes the lengths of three sides: **side1**, **side2** and **side3** of the triangle as the input from the user using **input** function and return the area of the triangle as the output. Also, assert that sum of the length of any two sides is greater than the third side.
5. Consider a showroom of electronic products, where there are various salesmen. Each salesman is given a commission of 5%, depending on the sales made per month. In case the sale done is less than 50000, then the salesman is not given any commission. Write a function to calculate total sales of a salesman in a month, commission and remarks for the salesman. Sales done by each salesman per week is to be provided as input. Assign remarks according to the following criteria**:**

Excellent: Sales >=80000

Good: Sales>=60000 and <80000

Average: Sales>=40000 and <60000

Work Hard: Sales < 40000

Write a Python function that takes a number as an input from the user and computes its factorial.

7. Write a Python function to return nth terms of Fibonacci sequence

8. Write a function that takes a number with two or more digits as an input and finds its reverse and computes the sum of its digits.

9. Write a function that takes two numbers as input parameters and returns their least common multiple and highest common factor.

10. Write a function that takes a number as an input and determine whether it is prime or not.

11. Write a function that finds the sum of the n terms of the following series:

a) $1 - x^2/2! + x^4/4! - x^6/6! + \ldots x^n/n!$

b) $1 + x^2/2! + x^4/4! + x^6/6! + \ldots x^n/n!$

12. Write a Python function that takes two strings as an input from the user and counts the number of matching characters in the given pair of strings.

13. Write a Python function that takes a string as an input from the user and displays its reverse.

14. Write a Python function that takes a string as an input from the user and determines whether it is palindrome or not.

15. Write a Python function to calculate the sum and product of two compatible matrices

16. Write a function that takes a list of numbers as input from the user and produces the corresponding cumulative list where each element in the list present at index i is the sum of elements at index j <= i.

17. Write a function that takes **n** as an input and creates a list of n lists such that ith list contains first five multiples of i.

18. Write a function that takes a sentence as input from the user and calculates the frequency of each letter. Use a variable of dictionary type to maintain the count.

19. Write a Python function that takes a dictionary of *word*:*meaning* pairs as an input from the user and creates an inverted dictionary of the form meaning:list-of-words.

20. Usage of Python debugger tool-pydb and PythonTutor.

21. Implementation of Linear and binary search techniques

22. Implementation of selection sort, insertion sort, and bubble sort techniques

23. Write a menu-driven program to create mathematical 3D objects Curve, Sphere, Cone, Arrow, Ring, and Cylinder.

24. Write a program that makes use of a function to accept a list of n integers and displays a histogram.

25. Write a program that makes use of a function to display sine, cosine, polynomial and exponential curves.

26. Write a program that makes use of a function to plot a graph of people with pulse rate p vs. height h. The values of p and h are to be entered by the user.

27. Write a function that reads a file **file1** and displays the number of words and the number of vowels in the file.

28. Write a Python function that copies the content of one file to another.

29. Write a function that reads a file **file1** and copies only alternative lines to another file **file2**. Alternative lines copied should be the odd numbered lin

BCA204: Computer System Architecture (Discipline Specific Core Course)
Credit: 06
Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)
Workload: 5 Lectures (Per Week), 1 Tutorial (Per Week)

Course Objective
This course introduces the students to the fundamental concepts of digital computer organization, Design and architecture. It aims to develop a basic understanding of the building blocks of the computer system and highlights how these blocks are organized together to architect a digital computer system.

Course Learning Outcomes
1. On successful completion of the course, students will be able to:
2. Design Combinational Circuits using basic building blocks. Simplify these circuits using Boolean algebra and Karnaugh maps. Differentiate between combinational circuits and sequential circuits.
3. Represent data in binary form, convert numeric data between different number systems and perform arithmetic operations in binary.
4. Determine various stages of instruction cycle and describe interrupts and their handling.
5. Explain how CPU communicates with memory and I/O devices.

Detailed Syllabus

UNIT-I: 10 mark
Digital Logic Circuits: Logic Gates, truth tables, Boolean algebra, digital circuits, Combinational circuits, sequential circuits, circuit simplification using Karnaugh map, Don't Care Conditions, flip-flops, characteristic tables

UNIT-II: 15 mark
Digital Components: Half Adder, Full Adder, Decoders, Multiplexers, Registers and Memory Units

UNIT-III: 10 mark
Data Representation and Basic Computer Arithmetic: Number system, complements, fixed and floating point representation. Alphanumeric representation. Addition, subtraction.

UNIT-IV: 15 mark
Register Transfer language: Register Transfer language, Register Transfer, Bus and Memory Transfers (Tree State Bus Buffers, Memory   Transfer), Arithmetic Micro-Operations, Logic Micro Operations, Shift Micro-   Operations, Arithmetic logical shift unit. Instruction codes, Computer registers, computer instructions, Timing and Control, Instruction cycle, Memory-Reference Instructions, Input output and interrupt, Design of Basic computer, design of Accumulator Unit.

UNIT-V: 15 mark
Central Processing Unit: Introduction, General Register Organization, Stack Organization, Instruction format, Addressing Modes, data transfer and manipulation, Program Control, Reduced Instruction Set Computer, CISC, Pipelining. Computer Arithmetic Introduction,

Addition and subtraction, Multiplication Algorithms (Booth Multiplication Algorithm), Division Algorithms, Floating Point Arithmetic operations, Decimal Arithmetic.

UNIT-VI:10 mark

Input-Output Interface, Asynchronous Data Transfer, Modes Of Transfer, Priority Interrupt, DMA, Input- Output Processor (IOP), CPU IOP Communication, Serial communication. Memory Organization: Memory Hierarchy, Main Memory, Auxiliary Memory, Associative Memory, Cache Memory, Virtual Memory

Text Books:
1. Mano, M. (1992). *Computer System Architecture*. 3rd edition. Pearson Education.

References:
1. Mano, M. (1995). *Digital Design*. Pearson Education Asia.
2. Null, L., & Lobur, J. (2018). *The Essentials of Computer Organization and Architecture*. 5th edition. (Reprint) Jones and Bartlett Learning.
3. Stallings, W. (2010). *Computer Organization and Architecture Designing for Performance* 8th edition. Prentice Hall of India.

BCA305: Data Structure using C (Discipline Specific Core Course)
Credit: 06
Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)
Workload: 4 Lectures (Per Week), 2 Practical (Per Week)

Course Objective

This course aims at developing the ability to use basic data structures like array, stacks, queues, lists, trees and hash tables to solve problems.

Course Learning Outcomes

At the end of the course, students will be able to:
1. Implement and empirically analyses linear and non-linear data structures like Arrays, Stacks, Queues, Lists, Trees, Heaps and Hash tables as abstract data structures.
2. Write a program, choosing a data structure, best suited for the application at hand.
3. Re-write a given program that uses one data structure, using a more appropriate/efficient data structure.
4. Write programs using recursion for simple problems. Explain the advantages and disadvantages of recursion.

Detailed Syllabus

UNIT-I: 10 mark

Arrays: single and multi-dimensional arrays, analysis of insert, delete and search operations in arrays (both linear search and binary search), implementing sparse matrices, and applications of arrays to sorting: selection sort, insertion sort, bubble sort, comparison of sorting techniques via empirical studies. Introduction to Vectors.

UNIT-II: 15 mark

Linked Lists: Singly- linked, doubly-linked and circular lists, analysis of insert, delete and search operations in all the three types, implementing sparse matrices. Introduction to Sequences.

UNIT-III: 10 mark
Queues: Array and linked representation of queue, de-queue, comparison of the operations on queues in the two representations. Applications of queues.

UNIT-IV: 15 mark
Stacks: Array and linked representation of stacks, comparison of the operations on stacks in the two representations, implementing multiple stacks in an array; applications of stacks: prefix, infix and postfix expressions, utility and conversion of these expressions from one to another; applications of stacks to recursion: developing recursive solutions to simple problems, advantages and limitations of recursion.

UNIT-V: 15 mark
Trees and Heaps: Introduction to tree as a data structure; binary trees, binary search trees, analysis of insert, delete, search operations, recursive and iterative traversals on binary search trees. Height-balanced trees (AVL), B trees, analysis of insert, delete, search operations on AVL and B trees. Introduction to heap as a data structure, Analysis of insert, extract-min/max and delete-min/max operations, applications to priority queues.

UNIT-VI: 10 mark
Hash Tables: Introduction to hashing, hash tables and hashing functions -insertion, resolving collision by open addressing, deletion, searching and their analysis, properties of a good hash function.

Practical
1. Write a program to search an element from a list. Give user the option to perform Linear or Binary search. Use Template functions.
2. WAP using templates to sort a list of elements. Give user the option to perform sorting using Insertion sort, Bubble sort or Selection sort.
3. 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 +).
4. Implement Doubly Linked List using templates. Include functions for insertion, deletion and search of a number, reverse the list.
5. Implement Circular Linked List using templates. Include functions for insertion, deletion and search of a number, reverse the list.
6. Perform Stack operations using Linked List implementation.
7. Perform Stack operations using Array implementation. Use Templates.
8. Perform Queues operations using Circular Array implementation. Use Templates.
9. Create and perform different operations on Double-ended Queues using Linked List implementation.
10. WAP to scan a polynomial using linked list and add two polynomial.
11. WAP to calculate factorial and to compute the factors of a given no. (i) using recursion, (ii) using iteration
12. WAP to display fibonacci series (i) using recursion, (ii) using iteration
13. WAP to calculate GCD of 2 number (i) with recursion (ii) without recursion

14. WAP to create a Binary Search Tree and include following operations in tree: (a) Insertion (Recursive and Iterative Implementation) (b) Deletion by copying (c) Deletion by Merging (d) Search a no. in BST (e) Display its preorder, postorder and inorder traversals Recursively (f) Display its preorder, postorder and inorder traversals Iteratively (g) Display its level-by-level traversals (h) Count the non-leaf nodes and leaf nodes (i) Display height of tree (j) Create a mirror image of tree (k) Check whether two BSTs are equal or not
15. WAP to convert the Sparse Matrix into non-zero form and vice-versa.
16. WAP to reverse the order of the elements in the stack using additional stack.
17. WAP to reverse the order of the elements in the stack using additional Queue.
18. WAP to implement Diagonal Matrix using one-dimensional array.
19. WAP to implement Lower Triangular Matrix using one-dimensional array.
20. WAP to implement Upper Triangular Matrix using one-dimensional array.
21. WAP to implement Symmetric Matrix using one-dimensional array.
22. WAP to create a Threaded Binary Tree as per inorder traversal, and implement operations like finding the successor / predecessor of an element, insert an element, inorder traversal.
23. WAP to implement various operations on AVL Tree.
24. WAP to implement heap operations.

Text book:
1. Drozdek, A., (2012), *Data Structures and algorithm in C++*. 3rd edition. Cengage Learning.
2. Goodrich, M., Tamassia, R., & Mount, D., (2011). *Data Structures and Algorithms Analysis in C++*. 2nd edition. Wiley.
3. G.S. Baluja ,Data Structure through C (A practical Approach) , Dhanpat Rai & Co.(p) LTD, New Delhi

References:
1. Foruzan, B.A. (2012) *Computer Science: A Structured Approach Using C++,* Cengage Learning
2. Lafore, R. (2008). *Object Oriented Programming in C++*. 4th edition. SAMS Publishing.
3. Sahni, S. (2011). *Data Structures, Algorithms and applications in C++*. *2nd*Edition, Universities Press
4. Tenenbaum, A. M., Augenstein, M. J., & Langsam Y., (2009), *Data Structures Using C and C++*. 2nd edition. PHI

BCA306: Operating system (Discipline Specific Core Course)
Credit: 06
Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)
Workload: 5 Lectures (Per Week), 1 Tutorial (Per Week)

Course Objective
The course introduces the students to different types of operating systems. Operating system modules such as memory management, process management and file management are covered in detail.

Course Learning Outcomes

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

1. Implement multiprogramming, multithreading concepts for a small operating system.
2. Create, delete, and synchronize processes for a small operating system.
3. Implement simple memory management techniques.
4. Implement CPU and disk scheduling algorithms.
5. Use services of modern operating system efficiently
6. Implement a basic file system.

Detailed Syllabus

UNIT-I: 15 mark

Introduction: Operating systems (OS) definition, Multiprogramming and Time Sharing operating systems, real time OS, Multiprocessor operating systems, Multicore operating systems, various computing environments.

UNIT-II: 15 mark

Operating System Structures: Operating Systems services, System calls and System programs, operating system architecture (Micro Kernel, client server) operating

UNIT-III: 15 mark

Process Management: Process concept, Operation on processes, Multi-threaded processes and models, Multicore systems, Process scheduling algorithms, Process synchronization. The Critical-section problem and deadlock characterization, deadlock handling.

UNIT-IV: 15 mark

Memory Management: Physical and Logical address space; Memory allocation strategies - Fixed and Variable Partitions, Paging, Segmentation, Demand Paging and virtual memory, Page Replacement algorithm.

UNIT-V: 15 mark

File and I/O Management: Directory structure, File access methods, Disk scheduling algorithms.

Practical

1. Write a program (using fork () and/or exec() commands) where parent and child execute: a) same program, same code. b) Same program, different code. - c) before terminating, the parent waits for the child to finish its task.
2. Write a program to report behavior of Linux kernel including kernel version, CPU type and model. (CPU information)
3. Write a program to report behavior of Linux kernel including information on 19 configured memory, amount of free and used memory. (Memory information)
4. Write a program to print file details including owner access permissions, file access time, where file name is given as argument.
5. Write a program to copy files using system calls.
6. Write a program to implement FCFS scheduling algorithm.
7. Write a program to implement Round Robin scheduling algorithm.
8. Write a program to implement SJF scheduling algorithm.
9. Write a program to implement non-preemptive priority based scheduling algorithm.
10. Write a program to implement preemptive priority based scheduling algorithm.

11. Write a program to implement SRJF scheduling algorithm.
12. Write a program to calculate sum of n numbers using thread library.
13. Write a program to implement first-fit, best-fit and worst-fit allocation strategies.

Text book:
1. Silberschatz, A., Galvin, P. B., & Gagne, G. (2008). *Operating Systems Concepts*. 8th edition.. John Wiley Publications.

References:
2. Dhamdhere, D. M. (2006). *Operating Systems: A Concept-based Approach*. 2nd edition. Tata McGraw-Hill Education.
3. Kernighan, B. W., & Rob Pike, R. (1984). *The Unix programming environment* (Vol. 270). Englewood Cliffs, NJ: Prentice-Hall
4. Stallings, W. (2018). *Operating Systems: Internals and Design Principles*. 9th edition. Pearson Education.
5. Tanenbaum, A. S. (2007). *Modern Operating Systems*. 3rd edition. Pearson Education.

BCA307: Discrete Mathematics (Discipline Specific Core Course)
Credit: 06
Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)
Workload: 5 Lectures (Per Week), 1 Tutorial (Per Week)

Course Objective The course aims to introduce the students to Boolean algebra, sets, relations, functions, principles of counting, and growth functions so that these concepts may be used effectively in other courses.

Course Learning Outcomes On successful completion of the course, students will be able to:
1. Define mathematical structures (relations, functions, sequences, series, and graphs) and use them to model real life situations.
2. Understand (trace) and construct simple mathematical proofs using logical arguments.
3. Solve class room puzzles based on counting principles.
4. Compare functions and relations with respect to their growth for large values of the input.

Detailed Syllabus

UNIT-I: 15 mark
Introduction: Sets - finite and infinite sets, uncountable infinite sets; functions, relations, properties of binary relations, closure, partial ordering relations; counting - Pigeonhole Principle, permutation and combination; mathematical induction, Principle of Inclusion and Exclusion.

UNIT-II: 15 mark

Growth of Functions: asymptotic notations, summation formulas and properties, bounding summations, approximation by integrals.

UNIT-III: 15 mark
Recurrence: recurrence relations, generating functions, linear recurrence relations with constant coefficients and their solution, recursion trees, Master Theorem

UNIT-IV: 15 mark
Graph Theory: basic terminology, models and types, multi-graphs and weighted graphs, graph representation, graph isomorphism, connectivity, Euler and Hamiltonian Paths and Circuits, planar graphs, graph coloring, Trees, basic terminology and properties of Trees, introduction to spanning trees.

UNIT-V: 15 mark
Propositional Logic: logical connectives, well-formed formulas, tautologies, equivalences, Inference Theory

Text book:
1. Mohapatra, & Liu, C. L. (2012). *Elements of Discrete mathematics*. 4th edition. McGraw Hill Education.
2. Rosen, K. H. (2011). *Discrete Mathematics and Its Applications*. 7th edition. Tata McGraw Hill Education.
3. C.L.Liu, Elements of Discrete Mathematics, Second Edition,Tata Mc-Graw-Hill.

References:
1. Albertson, M. O., & Hutchinson, J.P., (1988). *Discrete Mathematics with Algorithms.* John Wiley and Sons.
2. Cormen, T. H., Leiserson, C. E., & Rivest, R. L. (2009). *Introduction to algorithms*. 3rd edition. MIT Press.
3. Hein, J. L. (2015). *Discrete Structures, Logic, and Computability.* 4th edition. Jones and Bartlett Learning.
4. Hunter, D. J. (2011). *Essentials of Discrete Mathematics.* 2nd edition. Jones and Bartlett Learning

BCA408: Probability and Statistics (Discipline Specific Core Course)
Core Credit: 06
Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)
Workload: 5 Lecturers (Per Week), 2 Tutorials (Per Week)

Course Objectives: To make the students familiar with the basic statistical concepts and tools which are needed to study situations involving uncertainty or randomness. The course intends to render the students to several examples and exercises that blend their everyday experiences with their scientific interests.

Course Learning Outcomes: This course will enable the students to learn:
1. Distributions to study the joint behavior of two random variables.
2. To establish a formulation helping to predict one variable in terms of the other, i.e., correlation and linear regression.

3. Central limit theorem, which helps to understand the remarkable fact that the empirical frequencies of so many natural populations, exhibit a bell shaped curve.

Detailed Syllabus

UNIT-I: 15 mark
Probability Functions and Moment Generating Function: Sample space, Probability set function, Real random variables - Discrete and continuous, Cumulative distribution function, Probability mass/density functions, Transformations, Mathematical expectation, Moments, Moment generating function, Characteristic function.

UNIT-II: 20 mark
Univariate Discrete and Continuous Distributions: Discrete distributions**:** Uniform, Bernoulli, Binomial, Negative binomial, Geometric and Poisson; Continuous distributions**:** Uniform, Gamma, Exponential, Chi-square, Beta and normal; Normal approximation to the binomial distribution.

UNIT-III: 20 mark
Bivariate Distribution: Joint cumulative distribution function and its properties, Joint probability density function, Marginal distributions, Expectation of function of two random variables, Joint moment generating function, Conditional distributions and expectations.

UNIT-IV: 20 mark
Correlation, Regression and Central Limit Theorem: The Correlation coefficient, Covariance, Calculation of covariance from joint moment generating function, Independent random variables, Linear regression for two variables, The method of least squares, Bivariate normal distribution, Chebyshev's theorem, Strong law of large numbers, Central limit theorem and weak law of large numbers.

Text Books:
1. Hogg, Robert V., McKean, Joseph W., & Craig, Allen T. (2013). *Introduction to Mathematical Statistics* (7th ed.). Pearson Education, Inc.
2. Basic Statistics –Agarwal B.L, New age international, 6th edition, 2013
3. Miller, Irwin & Miller, Marylees. (2014). John E. Freund's *Mathematical Statistics wit Applications* (8th ed.). Pearson. Dorling Kindersley (India).

References:
1. Medhi J – Statistical methods, New age International,Second edition, Reprint 2013
2. Walpole, Myers et al. –Probability and statistics for scientists and engineers., Pearson Education, Ninth edition, 2013
3. Applied Statistics and probability for Engineers, Runger and Montgomery,Wiley, 6th Edition
4. Ross, Sheldon M. (2014). *Introduction to Probability Models* (11th ed.). Elsevier Inc. AP.

BCA409: Database Management Systems (Discipline Specific Core Course)
Credit: 06
Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)

Workload: 4 Lectures (Per Week), 4 Practical (Per Week)

Course Objective
The course introduces the foundations of database management systems focusing on significance of a database, relational data model, schema creation and normalization, transaction processing, indexing, and the relevant data structures (files and B+-trees).

Course Learning Outcomes
On successful completion of the course, students will:
1. Describe major components of DBMS and their function
2. Model an application's data requirements using conceptual modelling tools like ER diagramsand design database schemas based on the conceptual model.
3. Write queries in relational algebra / SQL
4. Normalize a given database schema to avoid data anomalies and data redundancy. Describe the notions of indexes, views, constraints and transactions.

Detailed Syllabus

UNIT-I: 10 mark
Introduction to databases: Characteristics of database approach, data models, database systemarchitecture, data independence and data abstraction.

UNIT-II: 10 mark
Data modeling: Entity relationship (ER) modeling: Entity types, relationships, constraints, ERdiagrams, EER model

UNIT-III:10 mark
Relation data model: Relational model concepts, relational constraints, relational algebra.

UNIT-IV: 15 mark
SQL queries: SQL data definition, data types, specifying constraints, Queries for retrieval, insertion, deletion, updation, introduction to views.

UNIT-V: 15 mark
Database design: Mapping ER/EER model to relational database, functional dependencies, Lossless decomposition, Normal forms (upto BCNF).

UNIT-VI: 15 mark
Transaction and data storage: Introduction to transaction processing: ACID properties, concurrency control; Introduction to indexing structures for files.

Text book:
1. Remez Elmasri and Shamkant B. Navathe, "Fundamentals of Database Systems", 5 th Edition, Pearson Education, 2007.
2. A.K. Majumdar, P. Bhattacharyya, Database Management Systems, Tata McGra-Hill, 1996
3. H. Korth, A.Silberschatz, Database System Concepts, McGraw-Hill (Second Edition), 1991
4. R. Elmasri, S. Navathe, Fundamentals of Database System, Benjamin Cummings (Second Edition), 1994

5. Bipin Desai, An Introduction to Database Systems, Galgotai Publication (West Publishing), 1991.

6. F. Mc-Fadden, J. Hoffer, Modern Database Management, Benjamin cummings (Narosa), (Fourth Edition), 1994

References:
   1. Abrahamsi. Silberschatz, Henry. F. Korth, S. Sudarshan, "Database System Concepts" 6th Edition, McGraw Hill, 2012. 2. C.J.Date, "Introduction to database systems", Eight Edition, Addison Wesley, 2003.
   2. Peter Rob, Carlos Coronel, Database Systems: Design, Implementation and Management, Wadsworth Publishing Company, 1993
   3. C.J. Date, An Introduction to Database Systems, Volume I, Addision Wesley (Fifth Edition), 1994 . J.D, Ullman, Principles of Database Systems. Galgotia Publishing (Second Edition), 1994. D.M. Kroenke, Database Processing : Fundamentals, Design Implementation, Prentice Hall (Fifth Edition) 1994

Practical (Database Management System)

Create the following Database
Salesman

SNUM	SNAME	CITY	COMMISSION
1001	RAM	LONDON	14%
1002	ANAND	NEWDELHI	10%
1003	SHYAM	MUMBAI	13%
1007	MOTILAL	NEWYORK	11%
1004	PIYUSH	BARODA	15%
1006	SEJAL	SPAIN	12%

Customer

CNUM	CNAME	CITY	RATING	SNUM
2001	Harsh	London	100	1001
2002	Gita	Rome	200	1003
2003	Lalit	Surat	200	1002
2004	Govind	Bombay	300	1002
2008	Chirag	London	100	1001
2006	Chinmay	Surat	400	1007

Orders

ONUM	AMOUNT	ODATE	CNUM	SNUM
3001	17.79	01/03/21	2009	1007
3003	676.91	01/03/21	2014	1006

3002	1800.21	01/03/21	2019	1004
3005	1650.54	01/03/21	2018	1002
3006	1808.61	01/03/21	2016	1005
3009	1719.32	10/04/21	2013	1008
3007	57.57	10/04/21	2017	1004
3008	7432.00	11/05/21	2020	1005
3010	3108.59	11/05/21	2012	1002
3011	8981.88	11/05/21	2011	1008

Practical List – 1

Solve the following queries using above databases and where clause range searching and pattern matching.

1. Produce the order no, amount and date of all orders.
2. Give all the information about all the customers with salesman number 1001.
3. Display the following information in the order of city, Sname, Snum and commission.
4. List of rating followed by the name of each customer in Surat.
5. List of snum of all salesmen with order in order table without any duplicates.
6. List of all orders for more than Rs.1000. List of names and cities of all salesmen in London with commission above 10% List all customers excluding those with rating <=100 unless they are located in London
7. List all orders for more than Rs.1000 except the orders of snum<1006 of 10/03/21.
8. List all order taken on October 3rd or 4th or 6th, 2008.
9. List all customers whose names begins with a letter ' A'.
10. List all customers whose names begins with letter 'A' to 'G'.
11. List all orders with zero or NULL amount.

Practical List – 2

1. Solve the following queries using above databases and where clause range searching and pattern matching
2. List all salesmen with their % of commission.
3. Display the no. Of orders for each day in the descending order of the no. Of orders in the following format. FOR dd-mm-yy,there are____ Orders.
    a. Assume each salesperson has a 12% commission. Write a query on the order table that will produce the order number, salesman no and the amount of commission for that order.
4. Find the highest rating in each city in the form: For the city (city), the highest rating)
5. List all in descending order of rating. Calculate the total of orders for each day and place the result in descending order.

BCA410: Web Technologies (Discipline Specific Core Course)
Credit 06
Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)
Workload: 4 Lectures (Per Week), 4 Practical (Per Week)

Course Objectives:
The course content enables students to:
1. Understand best technologies for solving web client/server problems

2. Analyze and design real time web applications
3. Use Java script for dynamic effects and to validate form input entry
4. Analyze to Use appropriate client-side or Server-side applications

Course Outcomes:
At the end of the course students are able to:
1. Choose, understand, and analyze any suitable real time web application.
2. To develop and deploy real time web applications in web servers and in the cloud.
3. Extend this knowledge to .Net platform

UNIT–I: 10 mark
Introduction to Web Design: Introduction of Internet, WWW, What is Website? How the Website Works?, Web pages, Front End, Back End, Client and Server Scripting Languages, Responsive Web Designing, Types of Websites (Static and Dynamic Websites) . Editors Notepad, Downloading free Editors: Notepad++, Sublime Text Editor, Making use of Editors, File creation and editing, saving

UNIT–II : 10 mark
HTML: Introduction, Basic Structure of HTML , Head Section and Elements of Head Section, Formatting Tags :Bold , Italic, Underline, Strikethrough, Div, Pre Tag Anchor links and Named Anchors ImageTag, Paragraphs, Comments, Tables : Attributes – (Border, Cellpadding, Cellspacing , height , width), TR, TH, TD, Rowspan, Colspan Lists : Ordered List , Unordered List, Definition List, Forms, Form Elements, Input types, Input Attriutes, Text Input Text Area, Dropdown Radio buttons , Check boxes ,Submit and Reset Buttons Frames : Frameset , nested Frames HTML 5 Introduction , HTML5 New Elements: Section , Nav, Article , Aside, Audio Tag, Video Tag, HTML5 Form Validations: Require Attribute, Pattern Attribute , Autofocus Attribute, email, number type, date type , Range type. Introduction to CSS, Types of CSS, CSS Selectors : Universal Selector ,ID , Tag Selector, Class Selector, Sub Selector, Attribute Selector, Group Selector, CSS Properties : Back Ground properties , Block Properties , Box properties , List properties , Border Properties , Positioning Properties, CSS Lists CSS Tables, CSS Menu Design CSS Image Gallery,

UNIT-III: 15 mark
Introduction to CSS: Types of CSS, CSS Selectors : Universal Selector ,ID , Tag Selector, Class Selector, Sub Selector, Attribute Selector, Group Selector,   CSS Properties : Back Ground properties , Block Properties , Box properties , List properties , Border Properties, Positioning Properties, CSS Lists CSS Tables, CSS Menu Design CSS Image Gallery.
Web Site Development using W3.CSS Framework, W3.CSS Intro, W3.CSS Colors, W3.CSS Containers, W3.CSS Panels, W3.CSSBorders, W3.CSSFonts, W3.CSS Text, W3.CSS Tables, W3.CSS List, W3.CSSImages, W3.CSS Grid

UNIT–IV: 15 mark
Javscript and Angular Js Introduction to Client Side Scripting Language, Variables in Java Script, Operators in JS, Conditions Statements, JS Popup Boxes, JS Events, Basic Form Validations in JavaScript.

 UNIT-V: 15 mark
 PHP Programming: Introducing PHP: Creating PHP script, Running PHP script. Working with Variables and constants: Using variables, Using constants, Data types, Operators.
Controlling program flow: Conditional statements, Control statements, Arrays, functions. Working with forms.

UNIT-VI: 15 mark
Web Publishing and Browsing Overview, SGML, Web hosting Basics, HTML, CGL, Documents Interchange Standards, Components of Web Publishing, Document management, Web Page Design Consideration and Principles, Search and Meta Search Engines, WWW, Browser, HTTP, Publishing Tools

Text Books:

1. ITL Education Solution Limited, Introduction to Information Technology, Pearson Education, 2012 DT Editorial Services, HTML 5 Black Book (Covers CSS3, JavaScript, XML, XHTML, AJAX, PHP, jQuery), Second Edition, Dreamtech Publisher, 2016
2. Web Technologies, Uttam Roy, OXFORD University press
3. Web programming with HTML, XHTML and CSS, 2e, Jon Duckett, Wiley Achyut Godbole,Atul Kahate, Web Technologies, Third Edition,Mc Graw Hill Education.
4. O level Web design and Publishing .

Reference Books:

1. Web programming Bai, Michael Ekedahl, CENAGE Learning, India edition.
2. An Introduction to Web Design + Programming, Paul S.Wang, India Edition Sharma &Sharma, "Developing E-Commerce Sites", Addison Wesley Ivan Bayross, "Web Technologies Part II", BPB Publications

Practical (Web Technologies)

1. Create an HTML file (e.g. first_page.html) that specifies a page that contains a heading and two paragraphs of text. As the texts in the heading and paragraphs you can use any texts you like.
2. Write a HTML program to design a form which should allow to enter your personal data (Hint: make use of text field, password field, e-mail, lists, radio buttons, checkboxes, submit button)
3. Write HTML Code to demonstrate the use of Anchor Tag for the Following:-
   a. Creating a web link that opens in a new window.
   b. Creating a web link that opens in the same window
   c. Reference within the same html document.
   d. Reference to some image.
   e. Making an image a hyperlink to display second image
4. Create an html page with following specifications. Title should be about my City. Place your City name at the top of the page in large text and in blue color. Add names of landmarks in your city each in a different color, style and typeface. One of the landmark, your college name should be blinking. Add scrolling text with a message of your choice
5. Create an html page with 7 separate lines in different colors. State color of each line in its text.
6. Create an html page containing the polynomial expression as follows : $a^0 + a^1x + a^2x^2 + a^3x^3$
7. Write a HTML code to generate following output

First Frame: Name and address		
Second frame Bulleted list of qualifications		Third frame Links to favourite sites
Fourth frame Scrolling message	Fifth frame Blinking reminders	Sixth frame Image

8. Create a HTML Document with JavaScript code that has three Textboxes and a button. The details should be accepted using textboxes are principal, rate of interest, and duration in years. When user clicks the OK Button a message box appears showing the simple interest of principal amount.
9. Write a HTML Script to insert a hyperlink. Create a hyperlink in html which when clicked links to www.google.com in a new window. Create a HTML file which displays three images at LEFT, RIGHT and CENTER respectively in the browser.
10. Create table with ROWSPAN and COLSPAN attribute of TABLE in HTML (Prepare timetable of your class). Include CELLSPACING & CELL PADDING.
11. Create a web page, divide the web page into four frames. In one frame create three links that will display different HTML forms in the remaining three frames respectively.
12. Write a program in Java Script to print factorial
13. With CSS use the shorthand background property to set background image to eg."xyz.png", show it once, in the top right corner.
14. Write a program in javascript to generate series of prime numbers. `Write a JavaScript program to display the current day and time in the following format.
    a. Sample Output: Today is: Tuesday.
    b. Current time is: 10 PM: 30:38
15. Write a program to sum and multiply of two numbers using JavaScript.
16. Write a program to redirect, popup and print function in JavaScript.
17. Create your first "Hello world" application in AngularJS.
18. Write a code to display the words "AngularJS" in both text format and in a text box when the page is viewed in the browser.
19. Create a sample form program that collects the first name, last name, email, user id, password and confirms password from the user. All the inputs are mandatory and email address entered should be in correct format. Also, the values entered in the password and confirm password textboxes should be the same. After validating using JavaScript, in output display proper error messages in red color just next to the textbox where there is an error.

BCA511: Computer Networks (Discipline Specific Core Course)
Credit: 06
Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)
Workload: 4 Lectures (Per Week), 4 Practical (Per Week)

Course Objective
This course covers the concepts of data communication and computer networks. It comprises of the study of the standard models for the layered protocol architecture to communicate between autonomous computers in a network and also the main features and issues of communication protocols for different layers. Topics covered comprise of introduction to OSI and TCP/IP models also.

Course Learning Outcomes
On successful completion of the course, the student will be able to:
1. Describe the hardware, software components of a network and their interrelations.
2. Compare OSI and TCP/IP network models.
3. Describe, analyze and compare different data link, network, and transport layer protocols.

4. Design/implement data link and network layer protocols in a simulated networking environment.

Detailed Syllabus

UNIT-I: 10 mark
Introduction: Types of computer networks, Internet, Intranet, Network topologies, Network classifications.
Network Architecture Models: Layered architecture approach, OSI Reference Model, TCP/IP Reference Model.

UNIT-II: 10 mark
Physical Layer: Analog signal, digital signal, digital modulation techniques (ASK, PSK, QAM), encoding techniques, maximum data rate of a channel, transmission media (guided transmission media, wireless transmission, satellite communication), multiplexing (frequency division multiplexing, time division multiplexing, wavelength division multiplexing).

UNIT-III: 15 mark
Data Link MAC Layer: Data link layer services, error-detection and correction techniques, error recovery protocols (stop and wait, go back n, selective repeat), multiple access protocols, (TDMA/FDP, CDMA/FDD/CSMA/CD, CSMA/CA), Data link and MAC addressing, Ethernet, data link layer switching, point-to-point protocol.

UNIT-IV: 15 mark
Network layer: Networks and Inter networks, virtual circuits and datagrams, addressing, sub netting, Routing- (Distance vector and link state routing), Network Layer Protocols- (ARP, IPV4, ICMP, IPV6).

UNIT-V: 15 mark
Transport and Application Layer: Process to process Delivery- (client server paradigm, connectionless versus connection oriented service, reliable versus unreliable); User Datagram Protocols, TCP/IP protocol, Flow Control.

UNIT-VI: 10 mark
Protocols: FTP (File Transfer protocol), SMTP (Simple, Mail Transfer Protocol), Telnet and remote login protocol, WWW (World Wide Web), HTTP (Hyper Text Transfer protocol), Uniform Resource Locator, HTML and forms.

Textbooks:
1. Forouzan, B. A. (2017). *Data Communication and Networking*. McGraw-Hill Education
2. Tanenbaum, A.S. & Wethrall,D.J. (2012). *Computer Networks*. Pearson Education

References:
1. Kozierok, C.M. *The TCP/IP Guide*, free online resource. (2005.). Retrieved from http://www.tcpipguide.com/free/index.htm
2. Kurose, J. F., & Ross, K. W. (2017). *Computer Networking: A Top-Down Approach*. Pearson Education India

3. Stallings, W. (2017). *Data and Computer Communications.* 10th edition. Pearson Education India.

Practical
1. Simulate Cyclic Redundancy Check (CRC) error detection algorithm for noisy channel.
2. Simulate and implement stop and wait protocol for noisy channel.
3. Simulate and implement go back n sliding window protocol.
4. Simulate and implement selective repeat sliding window protocol.
5. Simulate and implement distance vector routing algorithm
6. Simulate and implement Dijkstra algorithm for shortest path routing.

BCA512: Programming in JAVA (Discipline Specific Core Course)
Credit: 06
Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)
Workload: 4 Lectures (Per Week), 4 Practical (Per Week)

Course Objective
This course adds to the basic programming language skills acquired by the student in earlier semesters. The students are exposed to the advanced features available in Java such as exception handling, file handling, interfaces, packages and GUI programming.

Course Learning Outcomes
On successful completion of the course the student will be
1. Implement Exception Handling and File Handling.
2. Implement multiple inheritance using Interfaces.
3. Logically organize classes and interfaces using packages.
4. Use AWT and Swing to design GUI applications.

Detailed Syllabus

UNIT-I: 10 mark
Review of Object Oriented Programming and Java Fundamentals: Structure of Java programs, Classes and Objects, Data types, Type Casting, Looping Constructs.

UNIT-II: 10 mark
Interfaces Interface basics; Defining, implementing and extending interfaces; Implementing multiple inheritance using interfaces Packages Basics of packages, Creating and accessing packages, System packages, Creating user defined packages

UNIT-III: 10 mark
Exception handling using the main keywords of exception handling: try, catch, throw, throws and finally; Nested try, multiple catch statements, creating user defined exceptions

UNIT-IV: 15 mark
File Handling Byte Stream, Character Stream, File I/O Basics, File Operations

UNIT-V: 15 mark
AWT and Event Handling: The AWT class hierarchy, Events, Event sources, Event classes,

Event Listeners, Relationship between Event sources and Listeners, Delegation event model, Creating GUI applications using AWT.

UNIT-VI: 15 mark

S**wing**: Introduction to Swing, Swing vs. AWT, Hierarchy for Swing components, Creating GUI Applications using Swing.

Practical

1. Design a class Complex having a real part (x) and an imaginary part (y). Provide methods to perform the following on complex numbers:
   a. Add two complex numbers.
   b. Multiply two complex numbers.
   c. toString() method to display complex numbers in the form: $x + i y$
2. Create a class TwoDim which contains private members as x and y coordinates in package P1. Define the default constructor, a parameterized constructor and override toString() method to display the co-ordinates. Now reuse this class and in package P2 create another class ThreeDim, adding a new dimension as z as its private member. Define the constructors for the subclass and override toString() method in the subclass also. Write appropriate methods to show dynamic method dispatch. The main() function should be in a package P.
3. Define an abstract class Shape in package P1. Inherit two more classes: Rectangle in package P2 and Circle in package P3. Write a program to ask the user for the type of shape and then using the concept of dynamic method dispatch, display the area of the appropriate subclass. Also write appropriate methods to read the data. The main() function should not be in any package.
4. Create an exception subclass UnderAge, which prints "Under Age" along with the age value when an object of UnderAge class is printed in the catch statement. Write a class exceptionDemo in which the method test() throws UnderAge exception if the variable age passed to it as argument is less than 18. Write main() method also to show working of the program.
5. Write a program to implement stack. Use exception handling to manage underflow and overflow conditions
6. Write a program that copies content of one file to another. Pass the names of the files through command-line arguments.
7. Write a program to read a file and display only those lines that have the first two characters as '//' (Use try with resources).
8. Write a program to create an Applet. Create a frame as a child of applet. Implement mouseClicked( ), mouseEntered( ) and mouseExited( ) events for applet. Frame is visible when mouse enters applet window and hidden when mouse exits from the applet window.
9. Write a program to display a string in frame window with pink color as background.
10. Write a program to create an Applet that has two buttons named "Red" and "Blue". When button is pressed the background color of the applet is set to the color named by the button's label.
11. Create an applet which responds to KEY_TYPED event and updates the status window with message ("Typed character is: X"). Use adapter class for other two events.
12. Create an applet with two buttons labeled 'A' and 'B'. When button 'A' is pressed, it displays your personal information (Name, Course, Roll No, College) and when button 'B' is pressed, it displays your CGPA in previous semester.

13. Write a program that creates a Banner and then creates a thread to scrolls the message in banner from left to right across the applet's window. Rewrite the applet programs using Swing.

Text Books:
1. Balaguruswamy E. (2014). *Programming with JAVA: A Primer*. 5th edition. India: McGraw Hill Education
2. Schildt, H. (2018). *Java: The Complete Reference*. 10th edition. McGraw-Hill Education.

References:
1. Horstmann, C. S. (2017). *Core Java - Vol. I – Fundamentals* (Vol. 10). Pearson Education
2. Schildt, H., & Skrien, D. (2012). *Java Fundamentals - A Comprehensive Introduction*. India: McGraw Hill Education

BCA613: Theory of Computation (Discipline Specific Core Course)
Credit: 06
Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)
Workload: 4 Lecturers (Per Week), 1 Tutorial (Per Week)

Course Objective
This course introduces formal models of computation, namely, finite automaton, pushdown automaton, and Turing machine; and their relationships with formal languages. Students will also learn about the limitations of computing machines.

Course Learning Outcomes
On successful completion of the course, a student will be able to:
1. Design a finite automaton, pushdown automaton or a Turing machine for a problem at hand.
2. Apply pumping lemma to prove that a language is non-regular/non-context-free.
3. Describe limitations of a computing machine.

Detailed Syllabus

UNIT-I: 15 mark
Languages and Regular Expressions: Alphabets, string, language, basic operations on language, concatenation, union, Kleene star, Regular expressions.
.
UNIT-II: 15 mark
Regular Languages: Deterministic finite automata (DFA), Non-deterministic Finite Automata (NFA), relationship between NFA and DFA, Transition Graphs (TG), properties of regular languages, the relationship between regular languages and finite automata, Kleene's Theorem.

UNIT-III: 15 mark
Non-Regular Languages and Context Free Grammars: Pumping lemma for regular grammars, Context-Free Grammars (CFG),

UNIT-IV: 15 mark

Context-Free Languages (CFL) and PDA: Deterministic and non-deterministic Pushdown Automata (PDA), parse trees, leftmost derivation, pumping lemma for CFL, properties of CFL.

UNIT-V: 15 mark

Turing Machines and Models of Computations: Turing machine as a model of computation,
Configuration of simple Turing machine, Church Turing Thesis, Universal Turing Machine decidability, halting problem.

Textbooks:
1. Cohen, D. I. A. (2011). *Introduction to Computer Theory*. 2nd edition. Wiley India.
2. Lewis, H.R. & Papadimitriou, H. R. (2002). *Elements of the Theory of Computation*. 6th edition. Prentice Hall of India (PHI)

References:
1. Goodrich, M., Tamassia, R., & Mount, D.M. (2011). *Data Structures and Algorithms Analysis in C++*. 2nd edition. Wiley.
2. Gopalkrishnan, G.L. (2019) *Automata and Computability: A programmer's perspective*. CRC Press.
3. Linz, P. (2016). *An Introduction to Formal Languages and Automata*.6th edition. Jones and Bartlett Learning.

BCA614: Financial Accounting (Discipline Specific Core Course)
Credit: 06
Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)
Workload: 5 Lectures (Per Week), 1 tutorial (Per Week)

Objective of this course:
1. This course helps students to work with well-known Computerized accounting software i.e. Tally Prime
2. Student will learn to create company, enter accounting voucher entries including advance voucher entries, do reconcile bank statement, do accrual adjustments, and also print financial statements, etc. in Tally Prime software
3. Accounting with Tally certificate course is not just theoretical program, but it also includes continuous practice, to make students ready with required skill for employability in the job market

Outcome from this course:
1. After successfully qualifying practical examination, students will be awarded certificate to work with well-known accounting software i.e. Tally Prime Software.
2. Student will do by their own create company, enter accounting voucher entries including advance voucher entries, do reconcile bank statement, do accrual adjustments, and also print financial statements, etc. in Tally Prime software
3. Students do possess required skill and can also be employed as Tally data entry operator

Detailed Syllabus

UNIT–I: 15 mark
Meaning and Nature of Financial Accounting, Scope of Financial Accounting, Financial Accounting & Management Accounting, Accounting concepts & convention, Accounting Standards in India.

UNIT–II: 20 mark
Basis of accounting-cash & accrual, Journalizing transaction, Introduction to Ledger posting and trial balance, Capital and revenue items. Application of computers in accounting, accounting procedure used for recording cash, bank and journal transactions using appreciate vouchers, Introduction to ledger accounting, Cash Book, Journal and bank account, Introduction to trial balance, Profit and Loss account and balance sheet.

UNIT–III: 20 mark
Financial statement analysis: Ratio analysis, Funds flow analysis, concepts, uses, Preparation of funds flow statement, simple problem, Cash flow analysis, Concepts, uses, preparation of cash flow statement, simple problem, Break-even analysis.

UNIT–IV: 20 mark
Inventory valuation: Objectives, Introduction to FIFO, LIFO & Weighted Average method of inventory valuation, Valuation of inventory on balance sheet date, inventory accounting and control, Introduction to stocks & shares, Concept of cost of capital, introduction, importance, explicit & implicit cost, Measurement of cost of capital, cost of debt. Theory of working capital: Nature and concepts

TEXT BOOKS:
   1. Maheshwari & Maheshwari, "An Introduction to Accountancy", 8th Edition, Vikas Publishing House, 2003

REFERENCES BOOKS:
   2. Gupta R. L., Gupta V. K., "Principles & Practice of Accountancy", Sultan Chand & Sons, 1999.
   3. Khan & Jain, "Financial Accounting"
   4. Maheshwari S. N., "Principals of Management Accounting", 11th Edition, Sultan Chand & Sons, 2001.
   5. Shukla and Grewal, "Advanced Accounts", 14th Edition, Sultan Chand & Sons.

BCA715: Design and Analysis of Algorithms (Discipline Specific Core Course)
Credit: 06
Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)
Workload: 4 Lecturers (Per Week), 4 Practical (Per Week)

Course Objective

This course is designed to introduce the students to designed analysis of algorithms in terms of efficiency and correctness. The course focuses on highlighting difference between various problem solving techniques for efficient algorithm design.

Course Learning Outcomes:
On successful completion of this course, the student will be able to:
1. Given an algorithm, identify the problem it solves.
2. Write algorithms choosing the best one or a combination of two or more of the algorithm design techniques: Iterative, divide-n-conquer, Greedy, Dynamic Programming using appropriate data structures
3. Write proofs for correctness of algorithms.
4. Re-write a given algorithm replacing the (algorithm design) technique used with a more appropriate/efficient (algorithm design) technique.

Detailed Syllabus

UNIT-I: 15 mark
Algorithm Design Techniques: Iterative technique: Applications to Sorting and Searching (review), their correctness and analysis. Divide and Conquer: Application to Sorting and Searching (review of binary search), mergesort, quick sort, their correctness and analysis. Dynamic Programming**:** Application to various problems (for reference; Weighted Interval Scheduling, Sequence Alignment, Knapsack), their correctness and analysis. Greedy Algorithms: Application to various problems, their correctness and analysis.

UNIT-II: 15 mark
More on Sorting and Searching: Heapsort, Lower Bounds using decision trees, sorting in Linear Time -Bucket Sort, Radix Sort and Count Sort, Medians &Order Statistics, complexity analysis and their correctness.

UNIT-III: 15 mark
Advanced Analysis Technique: Amortized analysis, Accounting method or Taxation Method, potential Method.

UNIT-IV: 15 mark
Graphs: Graph Algorithms-Breadth First Search, Depth First Search and its Applications.

UNIT-V: 15 mark
Greedy method: General method, application-job sequencing with deadlines, Knapsack problem, Minimum cost spanning trees, single source shortest path problem.

Practical
1. Implement Insertion Sort (The program should report the number of comparisons)
2. Implement MergeSort(The program should report the number of comparisons)
3. Implement Heap Sort (The program should report the number of comparisons)
4. Implement Randomized Quick sort (The program should report the number of comparisons)
5. Implement Radix Sort
6. Create a Red-Black Tree and perform following operations on it:
    a. Insert a node

   b. Delete a node
   c. Search for a number &also report the color .of the node containing this number.
7. Write a program to determine the LCS of two given sequences
8. Implement Breadth-First Search in a graph
9. Implement Depth-First Search in a graph
10. Write a program to determine the minimum spanning tree of a graph.

Text Book
1. Kleinberg, J., &Tardos, E. (2013).*Algorithm Design*. 1st edition. Pearson Education India.

References
2. Cormen, T.H.,Leiserson,C.E. Rivest, R.L.,&Stein, C.(2015).*Introduction to Algorithms.*3rd edition. PHI.
3. Sarabasse&GlederA.V. (1999).*Computer Algorithm– Introduction to Design and Analysis*.3rd edition. Pearson Education

BCA716: Computer Graphics (Discipline Specific Core Course)
Credit: 06
Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)
Workload: 4 Lecturers (Per Week), 4 Practicals (Per Week)

Course Objective
This course introduces fundamental concepts of Computer Graphics with focus on modelling, rendering and interaction aspects of computer graphics. The course emphasizes the basic principles needed to design, use and understand computer graphics system.

Course Learning Outcomes
On successful completion of the course, students will be able to:
1. Describe Standard raster and vector scan devices as well as Graphical Input and output devices
2. Implement algorithms for drawing basic primitives such as line, circle and ellipse.
3. Implement algorithms for line clipping and polygon clipping and filling.
4. Implement a 3D object representation scheme and carryout 2D and 3D Transformation, 3D projections
5. Implement visible surface determination algorithms, Illumination models and surface rendering methods, color models
6. Implement a simple computer animation algorithm Detailed Syllabus

Detailed Syllabus

UNIT-I: 15 mark
I**ntroduction**: Introduction to Graphics systems, Basic elements of Computer graphics, Applications of computer graphics. Architecture of Raster and Random scan display devices, input/output devices.

UNIT-II: 15 mark
Drawing and clipping primitives: Raster scan line, circle and ellipse drawing algorithms, Polygon filling, line clipping and polygon clipping algorithms

UNIT-III: 15 mark
Transformation and Viewing: 2D and 3D Geometric Transformations, 2D and 3D Viewing Transformations (Projections- Parallel and Perspective), Vanishing points.

UNIT-IV: 15 mark
Geometric Modeling: Polygon Mesh Representation, Cubic Polynomial curves (Hermite and Bezier).

UNIT-V: 15 mark
Visible Surface determination and Surface Rendering: Z-buffer algorithm, List-priority algorithm and area subdivision algorithm for visible surface determination. Illumination and shading models, RGB color model and Basics of Computer Animation.

Practical
1. Write a program to implement Bresenham's line drawing algorithm.
2. Write a program to implement mid-point circle drawing algorithm.
3. Write a program to clip a line using Cohen and Sutherland line clipping algorithm.
4. Write a program to clip a polygon using Sutherland Hodgeman algorithm.
5. Write a program to fill a polygon using Scan line fill algorithm.
6. Write a program to apply various 2D transformations on a 2D object (use homogenous 64 Coordinates).
7. Write a program to apply various 3D transformations on a 3D object and then apply parallel and perspective projection on it.
8. Write a program to draw Hermite /Bezier curve.

Text Books:
1. Baker, D.H. (2008). Computer Graphics. 2nd edition. Prentice Hall of India.
2. Foley, J. D., Dam, A.V, Feiner, S. K., & Hughes, J. F. (1995). Computer Graphics: Principles and Practice in C. 2nd edition. Addison-Wesley Professional.

References:
1. Bhattacharya, S. (2018). Computer Graphics. Oxford University Press
2. Cohen, D. I. A. (2011). Introduction to Computer Theory. 2nd edition. Wiley India.
3. Marschner, S., & Shirley, P. (2017) Fundamentals of Computer Graphics. 4th edition. CRC Press
4. Rogers, D. F. (1989). Mathematical Elements for Computer Graphics. 2nd edition. McGraw Hill.

BCA817: Artificial Intelligence (Discipline Specific Core Course)
Credit: 06
Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)

Workload: 4 Lecturers (Per Week), 4 Practical (Per Week)

Course Objective
This course introduces the basic concepts and techniques of Artificial Intelligence (AI). The course aims to introduce intelligent agents and reasoning, heuristic search techniques, game-playing, knowledge representation, reasoning with uncertain knowledge.

Course Learning Outcomes
On successful completion of this course, students will be able to
1. Identify problems that are amenable to solution by specific AI methods
2. Represent knowledge in Prolog and write code for drawing inferences.
3. Identify appropriate AI technique for the problem at hand
4. Compare strengths and weaknesses of different artificial Intelligence techniques.
5. Sensitive towards development of responsible Artificial Intelligence

Detailed Syllabus

UNIT-I: 10 mark
Introduction: Introduction to artificial intelligence, background and applications, Turing test, rational agents, intelligent agents, structure, behavior and environment of intelligent agents.

UNIT-II: 15 mark
Knowledge Representation: Propositional logic, first order predicate logic, resolution principle, unification, semantic nets, conceptual dependencies, frames, scripts, production rules, conceptual graphs.

UNIT-III: 15 mark
Reasoning with Uncertain Knowledge: Uncertainty, non-monotonic reasoning, truth maintenance systems, default reasoning and closed world assumption, Introduction to probabilistic reasoning, Bayesian probabilistic inference, introduction to fuzzy sets and fuzzy logic, reasoning using fuzzy logic.

UNIT-IV: 15 mark
Problem Solving and Searching Techniques: Problem characteristics, production systems, control strategies, breadth first search, depth first search, hill climbing and its variations, heuristics search techniques: best first search, A* algorithm, constraint satisfaction problem, means-end analysis.

UNIT-V: 10 mark
Game Playing: introduction to game playing, min-max and alpha-beta pruning algorithms. Prolog Programming: Introduction to Programming in Logic (PROLOG), Lists, Operators, basic Input and Output.

UNIT-VI: 10 mark
Understanding Natural Languages: Overview of linguistics, Chomsky hierarchy of grammars, parsing techniques.

Practical
  1. Write a prolog program to calculate the sum of two numbers.

2. Write a Prolog program to implement max(X, Y, M) so that M is the maximum of two numbers X and Y.
3. Write a program in PROLOG to implement factorial (N, F) where F represents the factorial of a number N.
4. Write a program in PROLOG to implement generate_fib(N,T) where T represents the N term of the Fibonacci series.
5. Write a Prolog program to implement GCD of two numbers.
6. Write a Prolog program to implement power (Num,Pow, Ans) : where Num is raised to the power Pow to get Ans.
7. Prolog program to implement multi (N1, N2, R): where N1 and N2 denotes the numbers to be multiplied and R represents the result.
8. Write a Prolog program to implement memb(X, L): to check whether X is a member of L or not.
9. Write a Prolog program to implement conc (L1, L2, L3) where L2 is the list to be appended with L1 to get the resulted list L3.
10. Write a Prolog program to implement reverse (L, R) where List L is original and List R is reversed list.
11. Write a program in PROLOG to implement palindrome (L) which checks whether a list Lis a palindrome or not.
12. Write a Prolog program to implement sum list (L, S) that S is the sum of a given list L.
13. Write a Prolog program to implement two predicates even length (List) and odd length (List) so that they are true if their argument is a list of even or odd length respectively.
14. Write a Prolog program to implement nth_element (N, L, X) where N is the desired position, L is a list and X represents the Nth element of L.
15. Write a Prolog program to implement max list(L, M) so that M is the maximum number int he list.
16. Write a prolog program to implement insert_nth (I, N, L, R) that inserts an item I into N position of list L to generate a list R.
17. Write a Prolog program to implement delete_nth (N, L, R) that removes the element on Nth position from a list L to generate a list R.
18. Write a program in PROLOG to implement merge (L1, L2, L3) where L1 is first ordered list and L2 is second ordered list and L3 represents the merged list.

References
1. Rich, E. &Knight,K. (2012). Artificial Intelligence. 3rd edition. Tata McGraw Hill.
2. Russell, S.J. & Norvig, P. (2015) Artificial Intelligence - A Modern Approach. 3rd edition Pearson Education

Additional Resources:
1. Bratko, I. (2011). Prolog Programming for Artificial Intelligence. 4th edition. Pearson Education
2. Clocksin, W.F. & Mellish (2003), Programming in PROLOG. 5th edition. Springer
3. Kaushik, S. (2011). Artificial Intelligence. Cengage Learning India.
4. Patterson, D.W. (2015). Introduction to Artificial Intelligence and Expert Systems. 1st edition, Pearson Education.

Web Resources
1. https://cyber.harvard.edu/topics/ethics-and-governance-ai

2. https://royalsocietypublishing.org/doi/full/10.1098/rsta.2018.0085
3. https://arxiv.org/abs/1812.02953

BCA818: Software Engineering (Discipline Specific Core Course)
Credit 06
Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)
Workload: 5 Lectures (Per Week), 1 tutorial (Per Week)

Course Objectives:
The course objectives are to:
1. Acquire knowledge of basic software Engineering methods and practices, and their appropriate application.
2. Describe software engineering layered technology and process frame work.
3. Develop general understanding of software process models.
4. Understanding of software requirements and the SRS documents.
5. Understand different software architecture styles.
6. Understanding implementation issues such as modularity and coding standards.
7. Conceptualize quality issues and software evolution.

Course Outcomes:
On completion of the course students will be able to:
1. Gain basic knowledge and understanding of the analysis and design of complex systems.
2. Develop ability to apply software engineering principles and techniques.
3. Develop ability to develop, maintain and evaluate large – scale software systems.
4. Produce efficient, reliable, robust and cost-effective software solutions.
5. Manage time, processes and resources effectively.
6. Apply software testing and equality assurance techniques at the module level, and understand these techniques at the system and organization level.
7. Understanding common lifecycle processes
8. Prepare technical documentations and make presentations on various aspects of a software development project.

Detailed Syllabus

UNIT-I: 15 mark
Introduction: Software Crisis, Software Processes &Characteristics, Software life cycle Models, Waterfall, Prototype, Evolutionary and Spiral Models Software Requirements analysis & specifications: Requirement engineering, requirement elicitation techniques like FAST,QFD, requirements analysis using DFD, Data dictionaries &ER Diagrams, Requirements documentation, Nature of SRS, Characteristics & organization of SRS.

UNIT-II: 15 mark

Software Project Management Concepts: The Management spectrum, The People the Problem, The Process, The Project Software Project Planning: Size Estimation like lines of Code &Function Count, Cost Estimation Models, COCOMO, Risk Management

UNIT–III: 15 mark
Software Design: Cohesion& Coupling, Classification of Cohesiveness& Coupling, Function Oriented Design, Object Oriented Design Software Metrics: Software measurements: What &Why, Token Count,   Halstead Software Science Measures, Design Metrics, Data Structure Metrics,

UNIT–IV: 15 mark
Software Testing: Testing Process, Design of Test Cases, Types of Testing, Functional Testing, Structural Testing, Test Activities, Unit Testing, Integration Testing and System Testing. Debugging Activities

UNIT-V: 15 mark
Software Maintenance: Management of Maintenance, Maintenance Process, Revers Engineering, Software Reengineering, Configuration Management, Documentation.

Text Books:
1. K.K. Aggarwal& Yogesh Singh,"Software Engineering",2nd Ed.New Age International, 2005.
2. R.S.Pressman,"SoftwareEngineering–Apractitioner"sapproach",5thEd.,McGrawHill Int. Ed., 2001.

Reference Books:
1. Stephen R. Schach, "Classical & Object-Oriented SoftwareEngineering",IRWIN, 1996.
2. James Peter,W.Pedrycz,"Software Engineering: An Engineering Approach", John Wiley & ons, Somerville, "Software Engineering", Addison Wesley, 2002

Discipline Specific Elective Course- (DSE-1)

BCA519A: .NET Programming
Core Credit 06
Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)
Workload: 4 Lectures (Per Week), 4 Practical (Per Week)

Objective of the Course:
1. This course is designed to provide the knowledge of Dot Net Frameworks along with C#.
2. To explore .NET technologies for designing and developing dynamic, interactive and responsive web applications.
3. Provide a consistent, object-oriented programming environment whether object code is stored and executed locally, executed locally but web distributed, or executed remotely.
4. Build all communication on industry standards to ensure that code based on .NET Framework integrates with any other code.

Learning Outcomes:
1. After completion of the course the student will be able to use the features of Dot Net Framework along with the features of C#.
2. Use ADO.NET for data persistence in a web application.
3. To understand the 3-tier software architecture (presentation/client tier, application tier, data tier) and develop multi-tier applications to understand and experiment with the deployment of enterprise applications.

Detailed Syllabus

UNIT-I: 15 mark
I**ntroduction**: .Net architecture, Namespheres, Assemblies, object-oriented features, memory management, interoperation with 10M, transaction in .NET, structured exception handling, code access security.

UNIT-II: 15 mark
C# language, OOPs Concepts, Collections-Generic and Non-Generic, Database Programming using ADO.NET

UNIT-III: 15 mark
VB.NET Similarities & differences with Visual Basic, windows focus, ADO.NET, working with databases, object oriented features.

UNIT-IV: 15 mark
ASP.NET Similarities & difference with ASP, Architecture, web-form, development, XML, databases interface. C++.NET Similarities& differences with C/C++, Creating components, window four, menus, validation, database interface.

UNIT-V: 15 mark
Overview of Microsoft Database Access Technology, ADO.Net, Creating a Database, ADO.Net Architecture, ADO.Net Class Libraries, and Data bound Control.

Text Book:
1. Chakraborti et. al., "Microsoft .NET framework", PHI, 2002
2. M. Reynolds et. al., ".NET Enterprise", WroxlSPD, 2002

References:
1. Richard Blaur& Mathew Reynolds, "Beginning VB.net 2003", 3rd Edition, WileyDream Tech, 2003
2. Chris Willman,JohnKauffiman, Beginning ASP net 1.1 with VB. NET 2003",Wiley Dream Teach, 2003
3. Chris Ullman, John Kauffiman, Beginning ASP. NET with Visual # net 2003", Wiley Dream Teach, 2003.

Practical

1. Write console program in C#/VB.NET implementing the concept of OOP
2. Write console program in C# illustrating the concept of generic Collection using List, Dictionary, Stack, and Queue.
3. Write console program in C# illustrating the concept of non-generic Collection using Array List, Hash Table, Sorted List,Stack, Queue.
4. Develop window application using VB.NET.
5. Develop web application using C#/VB.NET.

BCA519B: Network Security (Discipline Specific Elective)
Credit: 06
Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)
Workload: 4 Lecturers (Per Week), 4 Practical's (Per Week)

Course Objective

The course offers abroad overview of the fundamentals of information security covering topics such as error correction/detection, cryptography, steganography, malwares, and this course also touches on the implications of security in Internet of Things (IoT).

Course Learning Outcomes

On successful completion of this course, a student will be able to
1. Identify the major types of threats to information security
2. Describe the role of cryptography in security
3. Select appropriate error-detection and error-correction methods for an application
4. Discuss the strengths and weaknesses of private and public key crypto systems
5. Describe malwares and memory exploits
6. Discuss the need for security in IOT,

Detailed Syllabus

Unit-I: 20 mark

Information security: Attributes of Information Security: Confidentiality, Integrity, and Availability. Threats& Vulnerabilities: Unauthorized Access, Impersonation, Denial of Service, Malicious Software; Trap Doors, Logic Bomb,Trojan Horses; Viruses,Worms& Bacteria;Cryptography Basics: Plain Text, Cipher Text, Encryption Algorithm, Decryption Algorithm; Requirements for Cryptography, Symmetric vs Asymmetric, Block and Stream ciphers, DES.

Unit–II: 20 mark

Public Key Infrastructure &. Message Authentication: Public Key Cryptography Principles &Applications, Algorithms: RSA, Message Authentication: One way Hash Functions: Message Digest, MD5, and SHA1. Public Key Infrastructure: Digital Signatures, Digital Certificates, Certificate Authorities.

Unit-III: 20 mark

Network Security: Network Attacks: Buffer Overflow, IP Spoofing, TCP Session Hijacking, Sequence Guessing, Network Scanning: ICMP,TCP sweeps, Basic Port Scans; Denial of Service Attacks :SYN Flood, Teardro pattacks, land, Smurf Attacks. IP security Architecture: Overview, Authentication header, Encapsulating Security Pay Load, combining Security Associations, Key Management. Virtual Private Network Technology: Tunneling using IPSEC.

Unit-IV: 15 mark

Web Security :Requirements, Secure Socket Layer and Secure Electronic Transactions, Network Management Security: Overview of SNMPArchitecture-SNMPV1, SNMPV3. Firewall Characteristics &Design Principles, Types of Firewalls: Packet Filtering Router, Application Level Gateway or Proxy, Content Filters, Bastion Host.

Practical

1. Implement the error correcting code.
2. Implement the error detecting code.
3. Implement caeser cipher substitution operation.
4. Implement mono alphabetic and polyalphabetic cipher substitution operation.
5. Implement play fair cipher substitution operation.
6. Implement hill cipher substitution operation.
7. Implement trail fence cipher transposition operation.
8. Implement row transposition cipher transposition operation.
9. Implement product cipher transposition operation.
10. Illustrate the Cipher text only and Known plaintext attacks.
11. Implement a stream Cipher technique

References

1. Pfleeger, C.P., Pfleeger, S.L.& Margulies, J. (2015).*Security in Computing*.5th edition. Prentice Hall
2. Lin, S. & Costello, D. J. (2004).*Error Control Coding: Fundamentals and applications*. 2nd edition. Pearson Education
3. Stallings, W. (2018). *Cryptography and network security*. 7th edition. Pearson Education.

Additional Resources

1. Berlekamp, E. R. (1986). *Algebraic Coding Theory*. McGraw Hill Book Company
2. Stallings, W. (2018) *Network security, essentials*.6th edition. Pearson Education.
3. Whitman M.E., & Mattord H.J. (2017).*Principle of Information Security*. 6th edition. Cengage Learning.

BCA519C: Computer Oriented Numerical Methods
Credit: 06
Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)
Workload: 4 Lectures (Per Week), 2 Practical (Per Week)

Course Objective: To enable students to
1. Obtain an intuitive and working understanding of numerical methods for the basic problems of numerical analysis.

2. Gain an experience in the implementation of numerical methods using a computer.
3. Trace error in these methods and need to analyze and predict it.
4. Have an adequate understanding of the application of Statistical methods.

Course learning outcome:

1. Understanding and Learning of numerical methods for numerical analysis.
2. Understanding the implementation of numerical methods using a computer.
3. Learning of tracing errors in Numerical methods and analyze and predict it.
4. Learning of application of Statistical methods.
5. Discuss concepts of numerical methods used for different applications.

Detailed Syllabus

UNIT-I:15 mark
Errors &Their Accuracy, Solutions of Algebraic and Transcendental Equations, Bisection Method, The method off alse position, the iteration method, Newton Raphson method, Generalized Newton's method, Solutions of system of non-linear Equations: The method of iteration, Newton Raphson method.

UNIT-II:20 mark
Interpolation Errors in polynomial interpolation, Finite Differences Detection of errors by use of difference tables, Differences of a Polynomial, Newton's formulae for interpolation, Central difference interpolation formulae, Guass's Central difference formulae, Interpolation with unevenly spaced points Lagrange's interpolation, Errorin Lagrange's interpolation, Divided differences and their properties, Newton's general interpolation formula-interpolation by iteration, inverse interpolation.

UNIT-III:20 mark
Differentiation and Integration, Numerical Differentiation, Methods based on interpolation, Nonuni form nodal points-Linear interpolation, and quadratic interpolation Uniform nodal points-Linear interpolation, quadratic interpolation Methods based on finite differences Numerical Integration Methods based on interpolation-Newton's Cotes methods, Trapezoidal method, Simpson's method Simpson's rule, open type integration rules. Methods based on undetermined coefficients-Newton's methods, Trapezoidal rule, Simpson's rule.

UNIT-IV:20 mark
Solution of System of linear equation by iteration method, Gauss-Sidel method, Jacobi's method, Numerical solution of Ordinary Differential Equation. Solution by Taylor's series, Euler's method, Modified Euler's method, Runge-Kutta method.

Books:
1. Introductory methods of Numerical Analysis-ByS. S. Sastry, PHI
2. Numerical Methods for Scientific and Engineering students-M.K.Jain, S.R.K.Iyengar, New age international (P)Ltd.
3. Computer Oriented Numerical Methods,-V. Rajaraman, PHI

Lab
1. To find out the root of the Algebraic and Transcendental equations using Bisection method.

2. To find out the root of the Algebraic and Transcendental equations using Newton-Raphson method.
3. To implement Numerical Integration using Trapezoidal rule.
4. To implement Numerical Integration using Simpson 1/3 rule.
5. To implement Numerical Integration Simpson3/8 rule.

Discipline Specific Elective Course- (DSE-II)

BCA620A: Internet of Things and its Applications (Discipline Specific Elective Course)
Credit: 06
Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)
Workload: 4 Lectures (Per Week), 4 Practical (Per Week)
Course Objective

The course aims to equip the students to understand the basics of Internet of Things (IoT) and its applications. IoT primarily refers to the connected and smarter world having physical and virtual objects with some unique identities. IoT applications spans across domains of industrial control, retail, energy, agriculture, etc.

Course Learning Outcomes
On successful completion of this course, the student will be able to
1. Understand how connected devices work together to update other applications.
2. Acquire knowledge to interface sensors and actuator with microcontroller based Arduino platform.
3. Writing C programs in Arduino IDE.
4. Build IoT based applications and understand how data flows between things.

Detailed Syllabus

UNIT-I:15 mark
Introduction - Overview of Internet of Things(IoT), the characteristics of devices and applications in IoT ecosystem, building blocks of IoT, Various technologies making up IoT ecosystem, IoT levels, IoT design methodology, The Physical Design/Logical Design of IoT, Functional blocks of IoT and Communication Models.

UNIT-II:15 mark
Working of Controlled Systems, Real-time systems with feedback loop e.g. thermostat in refrigrator, AC, etc. Connectivity models – TCPIP versus OSI model, different type of modes using wired and wireless methodology, the process flow of an IoT application.

UNIT-III: 15 mark
Sensor - Measuring physical quantities in digital world e.g. light sensor, moisture sensor, temperature sensor, etc.
Actuator – moving or controlling system e.g. DC motor, different type of actuators

Controller – Role of microcontroller as gateway to interfacing sensors and actuators, microcontroller vs microprocessor, different type of microcontrollers in embedded ecosystem.

UNIT-IV: 15 mark

Building IoT applications: Emedded 'C' Language basics - Variables and Identifiers, Built-in Data Types, Arithmetic operators and Expressions, Constants and Literals, assignment. Conditional Statements and Loops - Decision making using Relational Operators, Logical Connectives - conditions, if-else statement, Loops: while loop, do while, for loop, Nested loops, Infinite loops, Switch statement.

UNIT-V: 15 mark

Arrays – Declaring and manipulating single dimension arrays
Functions - Standard Library of C functions in Arduino IDE, Prototype of a function: Formal parameter list, Return Type, Function call.
Interfacing sensors – The working of digital versus analog pins in Arduino platform, interfacing LED, Button, Sensors-DHT, LDR, MQ135. Display the data on Liquid Crystal Display (LCD), interfacing keypad, Serial communication – interfacing HC-05(Bluetooth module)
Control/handle 220v AC supply – interfacing relay module.

Practical

For Practical students may use Arduino and sensors/actuators
1. Interfacing Light Emitting Diode(LED)- Blinking LED
2. Interfacing Button and LED – LED blinking/glow when button is pressed
3. Interfacing Light Dependent Resistor (LDR) and LED, displaying automatic night lamp
4. Interfacing Temperature Sensor(LM35) and/or humidity sensor (e.g. DHT11)
5. Interfacing Liquid Crystal Display(LCD) – display data generated by sensor on LCD
6. Interfacing Air Quality Sensor-pollution (e.g. MQ135) - display data on LCD, switch on LED when data sensed is higher than specified value.
7. Interfacing Bluetooth module (e.g. HC05)- receiving data from mobile phone on Arduino and display on LCD .
8. Interfacing Relay module to demonstrate Bluetooth based home automation application. (Using Bluetooth and relay).

Text Books:
1. O level IOT and its Application bpb Publication.
2. Internet Of Things: A Hands-On Approach by ArsheepBahga (Author), Vijay Madisetti (Author)
3. Internet Of Things: Key Applications And Protocols by Olivier Hersent and David Boswarthick, John Wiley
4. Internet of Things (IoT) A Quick Start Guide Chitra Lele (BPB Publication)
5. IOT (Internet of things) and Its Application _ O LEVEL Book Multilingual Edition By P K PANDEY

BCA620B: Cloud Computing (Discipline Specific Core Course)
Credit: 06
Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)

Workload: 5 Lectures (Per Week), 1 Tutorial (Per Week)
Course Objective:

In this on-line course we plan to give students an overview of the field of Cloud Computing, and an in-depth study into its enabling technologies and main building blocks. Students will gain hands-on experience solving relevant problems through projects that will utilize existing public cloud tools. It is our objective that students will develop the skills needed to become a practitioner or carry out research projects in this domain. Specifically, the course has the following objectives: Students will learn

1. The fundamental ideas behind Cloud Computing, the evolution of the paradigm, its applicability; benefits, as well as current and future challenges.
2. The basic ideas and principles in data center design; cloud management techniques and cloud software deployment considerations.
3. different CPU, memory and I/O virtualization techniques that serve in offering software, computation and storage services on the cloud; Software Defined Networks (SDN) and Software Defined Storage (SDS).
4. Cloud storage technologies and relevant distributed file systems, NoSQL databases and object storage.
5. The variety of programming models and develop working experience in several of them.

Learning Outcomes:

The primary learning outcomes of this course are five-fold. Students will be able to:

1. Explain the core concepts of the cloud computing paradigm: how and why this paradigm shift came about, the characteristics, advantages and challenges brought about by the various models and services in cloud computing.
2. Apply fundamental concepts in cloud infrastructures to understand the tradeoffs in power, efficiency and cost, and then study how to leverage and manage single and multiple datacenters to build and deploy cloud applications that are resilient, elastic and cost-efficient.
3. Discuss system, network and storage virtualization and outline their role in enabling the cloud computing system model.
4. Illustrate the fundamental concepts of cloud storage and demonstrate their use in storage systems such as Amazon S3 and HDFS.
5. Analyze various cloud programming models and apply them to solve problems on the cloud.

Detailed Syllabus

UNIT-I: 15 mark

Origins of Cloud computing–Cloud components- Essential characteristics–On-demand self-service, Broad network access, Location independent resource pooling, Rapid elasticity, Measured service, Comparing cloud providers with traditional IT service providers, Roots of cloud computing. Cloud Consumer, Cloud Service Owner, Cloud Characteristics, On-Demand Usage, Ubiquitous Access Multitenancy (and Resource Pooling),Elasticity, Measured Usage, Resiliency Architectural influences–High-performance computing, Utility and Enterprise grid computing, Cloud scenarios–Benefits: scalability, simplicity, vendors, security, Limitations– Sensitive information - Application development- security level of third party - security benefits, Regularity issues: Government policies.

UNIT-II: 20 mark

Layers in cloud architecture, Software as a Service(SaaS),features of SaaS and benefits, Platform as a Service (PaaS), features of PaaS and benefits, Infrastructure as a Service (IaaS), features of IaaS and benefits, Service providers, challenges and risks in cloud adoption. Cloud deployment model: Public clouds–Private clouds–Community clouds- Hybrid clouds-Advantages of Cloud computing.

UNIT-III: 20 mark

Introduction, Storage as a service, Amazon storage services, Compute as a service Amazon elastic compute cloud(EC2),Cloud System matrix, Platforms Service, Windows Azure, Google Apps Engine, Amazon Web services, Software asa Service CRM as a service, sales force.

Introduction to Data center, Virtualization, Standardization and modularity, Automation, Remote operation and management, Data center Security and facilities, Computing hardware, storage hardware, Network hardware, LAN fabric, SANfabric, NAS gateways.

UNIT-IV : 20 mark

Server Virtualization, Hypervisor based Virtualization, Hardware support Virtualization, VMware Virtualization software, Storage Virtualization, Hardware independence, Server Consolidation, Resource replication, Virtualization Management, Hypervisor clustering architecture. Defining Mobile Market, connecting to the cloud, Adopting mobile cloud application, Smart phones with the Cloud, Android, Apple iPhone, Blackberry, Symbian, Windows mobile, Mobile web service, Mobile interoperability, Location awareness, Push Service, Using SMS, Defining WAP and other Protocol, Performing Synchronization

Text Book:
1. Cloud Computing: Concepts, Technology & Architecture By Thomas Erl, Ricardo
2. Cloud computing a practical approach-AnthonyT.Velte,TobyJ.VelteRobertElsenpeter, TATA McGraw-Hill, New Delhi–2010
3. Cloud Computing: Web-Based Applications That Change the Way You Work and Collaborate Online -Michael Miller-Que2008

References:
1. Moving to Cloud by Dinkar Sitaram, Geetha Manjunath, Publication: Syngress Elsevier Inc, 2014(2ndEdition)
2. Cloud Computing Second Edition by Dr Kumar Saurabh, Publication WillyINDAI (2013)
3. Cloud Computing Bible by Barrie Sosinsky, Publisher Willy INDAI (2014)
4. Cloud computing for dummies-Judith Hurwitz, Robin Bloor, Marcia Kaufman, Fern Halper, Wiley Publishing, Inc, 2010
5. CloudComputing(PrinciplesandParadigms),EditedbyRajkumarBuyya,JamesBroberg,Andrzej Goscinski, John Wiley&Sons, Inc. 2011

BCA620C: Machine Learning (Discipline Specific Elective)
Credit: 06
Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)

<div align="center">

Workload: 4 Lectures (Per Week), 4 Practical (Per Week)
Course Objective
</div>

The course aims at introducing the basic concepts and techniques of machine learning so that a student can apply machine learning techniques to a problem at hand.

Course Learning Outcomes
On successful completion of this course, the student will be able to:
1. Differentiate between supervised and unsupervised learning tasks.
2. Differentiate between linear and non-linear classifiers.
3. Describe theoretical basis of SVM
4. Implement various machine learning algorithms learnt in the course.

Detailed Syllabus

UNIT-I: 10 mark
Introduction: Basic definitions, Hypothesis space and inductive bias, Bayes optimal classifier and Bayes error, Occamsrazor, Curse of dimensionality, dimensionality reduction, feature scaling, feature selection methods.

UNIT-II: 15 mark
Regression: Linear regression with one variable, linear regression with multiple variables, gradient descent, logistic regression, over-fitting, regularization. Performance evaluation metrics, validation methods

UNIT-III: 25 mark
Classification: Decision trees, Naive Bayes classifier, k-nearest neighbor classifier, perceptron, multilayer perceptron, neural networks, back-propagation algorithm, Support Vector Machine (SVM), Kernel functions

UNIT-IV: 25 mark
Clustering: Approaches for clustering, distance metrics, K-means clustering, expectation maximization, hierarchical clustering, performance evaluation metrics, validation methods.

Practical
For practical Labs for Machine Learning, students may use software's like MABLAB/Octaveor
Python. For later exercises, students can create/use their own datasets or utilize datas ets from online repositories like UCIMachine Learning Repository(http://archive.ics.uci.edu/ml/).
1. Perform elementary mathematical operations in Octave/MATLAB like addition, multiplication, division and exponentiation.
2. Perform elementary logical operations in Octave/MATLAB (like OR, AND, Checking for Equality, NOT, XOR).
3. Create, initialize and display simple variables and simples strings and use simple formatting for variable.
4. Create/Define single dimension / multi-dimension arrays, and arrays with specific values like array of all ones, all zeros, array with random values within arrange, or a diagonal matrix.
5. Use command to compute the size of a matrix, size/length of a particular row/column, load data from a text file, store matrix data to a text file, finding out variables and their features in the current scope

6. Perform basic operations on matrices (like addition, subtraction, multiplication) and display specific rows or columns of the matrix.
7. Perform the matrix operations like converting matrix data to absolute values, taking the negative of matrix values, add/removing rows/columns from a matrix, finding the maximum or minimum values in a matrix or in a row/column, and finding the sum of some/all elements in a matrix.
8. Create various type of plots/charts like histograms, plot based on sine/cosine function based on data from a matrix. Further label different axes in a plot and data in a plot.
9. Generate different subplots from a given plot and color plot data.
10. Use conditional statements and different type of loops based on simple example/s.
11. Perform vectored implementation of simple matrix operation like finding the transpose of a matrix, adding, subtracting or multiplying two matrices.
12. Implement Linear Regression problem. For example, based on a data set comprising of existing set of prices and area/size of the houses, predict the estimated price of a given house.
13. Based on multiple features/variables perform Linear Regression. For example, based on a number of additional features like number of bedrooms, servant room, number of balconies, Number of houses of years a house has been built– predict the price of a house.
14. Implement a classification/ logistic regression problem. For example based on different features of students data, classify, whether a student is suitable for a particular activity. Based on the available dataset, a student can also implement another classification problem like checking whether a mail is spam or not.
15. Use some function for regularization of dataset based on problem 14.
16. Use some function for neural networks, like Stochastic Gradient Descent or back propagation algorithm to predict the value of a variable based on the dataset of problem 14.

References
1. Flach, P. (2015).*Machine learning: The Art and Science of Algorithms that Make Sense of Data*. Cambridge University Press.
2. Mitchell, T.M. (2017).*Machine Learning*. McGraw Hill Education.

Additional References:
1. Christopher&Bishop, M. (2016).*Pattern Recognition and Machine Learning*. New York: Springer-Verlag
2. Haykins, S.O. (2010).*Neural Networks and Learning Machines*. 3rd edition. PHI.

Discipline Specific Elective Course- (DSE-III)

BCA741A: Deep Learning (Discipline Specific Elective)
Credit: 06
Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)
Workload: 4 Lectures (Per Week), 4 Practical (Per Week)

Course Objective
The objective of this course is to introduce students to deep learning algorithms and their applications in order to solve real problems.

Course Learning Outcomes

On successful completion of this course, the student will be able to:
1. Describe the feed-forward and deep networks
2. Design single and multi-layer feed-forward deep networks and tune various hyper parameters.
3. Implement deep neural networks to solve a problem
4. Analyse performance of deep networks.

Detailed Syllabus

UNIT-I: 15 mark
Introduction: Historical context and motivation for deep learning; basic supervised classification task, optimizing logistic classifier using gradient descent, stochastic gradient descent, momentum, and adaptive sub-gradient method.

UNIT-II: 10 mark
Neural Networks: Feed forward neural networks, deep networks, regularizing a deep network, model exploration, and hyper parameter tuning.

UNIT-III: 15 mark
Convolution Neural Networks: Introduction to convolution neural networks: stacking, striding and pooling, applications like image, and text classification.

UNIT-IV: 15 mark
Sequence Modelling: Recurrent Nets: Unfolding computational graphs, recurrent neural networks (RNNs), bidirectional RNNs, encoder-decoder sequence to sequence architectures, deep recurrent networks, LSTM networks.

UNIT-V: 15 mark
Auto encoders: Under complete auto encoders, regularized auto encoders, sparse auto encoders, DE noising auto encoders, representational power, layer, size, and depth of auto encoders, stochastic encoders and decoders.

UNIT-VI: 15 mark
Structuring Machine Learning Projects: Orthogonalization, evaluation metrics, train/dev/test distributions, size of the dev and test sets, cleaning up incorrectly labelled data, bias and variance with mismatched data distributions, transfer learning, multi-task learning.

Practical
1. Implement logistic regression classification with (a) gradient descent and (b) stochastic

   gradient descent method. Plot cost function over iteration.
2. Experiment with logistic regression by adding momentum term, and adaptive subgradient method
3. Write the code to learn weights of a perceptron for Boolean functions (NOT, OR, AND,

   NOR, and NAND).

4. Implement a feed-forward neural network for solving (a) regression and (b) 2-class classification problem. Also experiment with hyper-parameter tuning.
5. Train and test a feed-forward neural network for multi-class classification using soft max
layer as output
6. Create a 2D and 3D CNN for image classification. Experiment with different depth of network, striding and pooling values.
7. Implement (a) RNN for image classification, (b) GRU network and (c) Implement LSTM
network.
8. Implement an auto-encoder, denoising auto encoders and sparse auto encoders
9. Design a stochastic encoders and decoders.

References:
1. Bunduma, N. (2017). *Fundamentals of Deep Learning*. O'reilly Books.
2. Heaton, J.(2015). *Deep Learning and Neural Networks*, Heaton Research Inc.

Additional References:
1. Good fellow, I. (2016). *Deep Learning*. MIT Press.
2. Deng, L., & Yu, D. (2009). *Deep Learning: Methods and Applications (Foundations and Trends in Signal Processing)*. Publishers Inc.
3. Hall, M.L, (2011). *Deep Learning*. VDM Verlag

BCA721B: Data Analysis and Visualization (Discipline Specific Elective)
Credit: 06
Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)
Workload: 4 Lectures (Per Week), 4 Practical (Per Week)

Course Objective
This course introduces students to data analysis and visualization in the field of exploratory data science using Python.

Course Learning Outcomes
On successful completion of the course, the students will be able to:
1. Use data analysis tools in the pandas library.
2. Load, clean, transform, merge and reshape data.
3. Create informative visualization and summarize data sets.
4. Analyze and manipulate time series data.
5. Solve real world data analysis problems.

Detailed Syllabus

UNIT-I : 15 mark
Introduction: Introduction to Data Science, Exploratory Data Analysis and Data Science Process. Motivation for using Python for Data Analysis, Introduction of Python shell Python and Jupyter Notebook. **Essential Python Libraries:** NumPy, pandas, matplotlib, SciPy, scikit-learn, stats models

UNIT-II: 15 mark
Getting Started with Pandas: Arrays and vector zed computation, Introduction to pandas Data
Structures, Essential Functionality, Summarizing and Computing Descriptive Statistics. Data Loading, Storage and File Formats. Reading and Writing Data in Text Format, Web Scraping, Binary Data Formats, Interacting with Web APIs, Interacting with Databases Data Cleaning and Preparation. Handling Missing Data, Data Transformation, String Manipulation

UNIT-III: 15 mark
Data Wrangling: Hierarchical Indexing, Combining and Merging Data Sets Reshaping and Pivoting. **Data Visualization matplotlib**: Basics of matplotlib, plotting with pandas and sea born, other python visualization tools

UNIT-IV: 15 mark
Data Aggregation and Group operations: Group by Mechanics, Data aggregation, General split-apply-combine, Pivot tables and cross tabulation **Time Series Data Analysis**: Date and Time Data Types and Tools, Time series Basics, date Ranges, Frequencies and Shifting, Time Zone Handling, Periods and Periods Arithmetic, Resampling and Frequency conversion, Moving Window Functions.

UNIT-V: 15 mark
Advanced Pandas: Categorical Data, Advanced Group by Use, Techniques for Method Chaining

Practical
Use data set of your choice from Open Data Portal (https://data.gov.in/) for the following exercises.
1. Practical's based on Num Pynd array
2. Practical's based on Pandas Data Structures
3. Practicals based on Data Loading, Storage and File Format
4. Practicals based on Interacting with Web APIs
5. Practicals based on Data Cleaning and Preparation
6. Practicals based on Data Wrangling
7. Practicals based on Data Visualization using matplotlib
8. Practicals based on Data Aggregation
9. Practicals based on Time Series Data Analysis

References
1. McKinney, W.(2017).*Python for Data Analysis: Data Wrangling with Pandas*, *NumPy and IPython*. 2nd edition. O'ReillyMedia.
2. O'Neil, C., & Schutt,R. (2013).*Doing Data Science: Straight Talk from the Frontline* O'ReillyMedia

BCA721C: Fundamentals of Data Sciences
Credit: 06
Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)
Workload: 4 Lectures (Per Week), 4 Practical (Per Week)

Learning Objectives:
1. The goal of data science is to construct the means for extracting business-focused insights from data.
2. This requires an understanding of how value and information flows in a business, and
3. The ability to use that understanding to identify business opportunities.

Learning Outcomes:
1. Students will demonstrate proficiency with statistical analysis of data.
2. Students will develop the ability to build and assess data-based models.
3. Students will execute statistical analyses with professional statistical software.
4. Students will demonstrate skill in data management.

Detailed Syllabus

UNIT-I: 15 mark
Data Scientist's Tool Box: Turning data into actionable knowledge, introduction to the tools that will be used in building data analysis software: version control, mark down, git, GitHub, R, and R Studio.

UNIT-II: 15 mark
R Programming Basics: Overview of R, R data types and objects, reading and writing data, Control structures, functions, scoping rules, dates and times, Loop functions, debugging tools, Simulation, code profiling.

UNIT-III: 15 mark
Getting and Cleaning Data: Obtaining data from the web, from APIs, from databases and from colleagues in various formats. Basics of data cleaning and making data "tidy".

UNIT-IV: 15 mark
Exploratory Data Analysis: Essential exploratory techniques for summarizing data, applied before formal modeling commences, eliminating or sharpening potential hypotheses about the world that can be addressed by the data, common multivariate statistical techniques used to visualize high dimensional data.

UNIT-V: 15 mark
Reproducible Research: Concepts and tools behind reporting modern data analyses in a reproducible manner, to write a document using R markdown, integrate live R code into alliterate statistical program, compile R markdown documents using knitr and related tools, and organizea data analysis so that it is reproducible and accessible to others.

Reference Books
1. Rachel Schutt, CathyO'Neil, "Doing Data Science: Straight Talk from the Frontline" by Schroff/O'Reilly, 2013
2. Foster Provost, Tom Fawcett," Data Science for Business" What You Need to Know About Data Miningand Data-AnalyticThinking"byO'Reilly,2013.
3. JohnW. Foreman,"Data Smart: Using data Scienceto Transform Information into Insight"by JohnWiley&Sons, 2013.
4. Ian Ayres, "Super Crunchers: Why Thinking-by-Numbers Is the new Wayto Be Smart"Ist Edition by Bantam, 2007.

5. Eric Seigel, "Predictive Analytics: The Power to Predict who Will Click, Buy,Lie, or Die", 1 Edition, by Wiley, 2013.
6. Matthew A. Russel, "Miningthe Social Web: Data mining Facebook, Twitter,Linkedln, Goole+, GitHub, and More", Second Edition, byO'ReillyMedia, 2013.

Practical

1. Write program that prints 'Hello World 'to the screen.
2. Write program that asks the user for a number and prints the sum of the numbers 1 to n
3. Write program that prints a multiplication table for numbers up to 12.
4. Write function that returns the largest element in a list.
5. Write function that computes the running total of a list.
6. Write function that tests whether a string is a palindrome.
7. Implement linear search.
8. Implement binary search.
9. Implement matrices addition , subtraction and Multiplication
10. Fifteen students were enrolled in a course. Their ages were: 20 20 20 20 20 21 21 21 22 22 22 22 23 23 23
    a. Find the median age of all students under 22 years
    b. Find the median age of all students
    c. Find the mean age of all student
    d. Find the modal age for all students
    e. Two more students enter the class. The age of both students is 23. What is now mean, mode and median?
11. Following table gives a frequency distribution of systolic blood pressure. Compute all the measures of dispersion.

Midpoint	95.5	105.5	115.5	125.5	135.5	145.5	155.5	165.5	175.5
Number	5	8	22	27	17	9	5	5	2

12. Obtain probability distribution of , where X is number of spots showing when a six-sided Symmetric die (i.e. all six faces of the die are equally likely) is rolled. Simulate random samples of sizes 40, 70 and 100 respectively and verify the frequency interpretation of probability
13. Make visual representations of data using the base, lattice, and gg plot 2 plotting systems in R, apply basic principles of data graphics to create rich analytic graphics from available datasets.
14. UseGit /Github software to create Github account. Also, create are pousing Github

Discipline Specific Elective Course- (DSE-1V)

BCA822A: Digital Image Processing (Discipline Specific Core Course)
Credit: 06
Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)
Workload: 4 Lecturers (Per Week), 4 Practicals (Per Week)

Course Objective

This course introduces students to the fundamentals of digital image processing, and various image transforms, image restoration techniques, image compression and segmentation used in digital image processing.

Course Learning Outcomes
On successful completion of the course, students will be able to:
1. Describe the roles of image processing systems in a variety of applications
2. Write programs to read/write and manipulate images: enhancement, segmentation, and compression, spatial filtering.
3. Develop Fourier transform for image processing in frequency domain.
4. Evaluate the methodologies for image segmentation, restoration

Detailed Syllabus

Unit I: 15 mark
Introduction: Digital Image Fundamentals: Brightness, Adaptation and Discrimination, Light and Electromagnetic Spectrum, Image Sampling and Quantization, Some Basic Relationships between Pixels Types of images.

Unit II: 15 mark
Spatial Domain Filtering: Some Basic Intensity Transformation Functions, Histogram Equalization, Spatial Correlation and Convolution, Smoothening Spatial Filters: Low pass filters, Order Statistics filters; Sharpening Spatial Filters: Laplacian filter

Unit III: 10 mark
Filtering in Frequency Domain: The Discrete Fourier Transformation (DFT), Frequency Domain Filtering: Ideal and Butterworth Low pass and High pass filters, DCT Transform (1D,2D).

Unit IV: 15 mark
Image Restoration: Image Degradation/Restoration Process, Noise models, Noise Restoration
Filters **Image Compression:** Fundamentals of Image Compression, Huffman Coding, Run Length Coding, JPEG.

Unit 5: 10 mark
Morphological Image Processing: Erosion, Dilation, Opening, Closing, Hit-or-Miss Transformation, Basic Morphological Algorithms.

Unit 6: 10 mark
Image Segmentation: Point, Line and Edge Detection, Thresholding, Region Based Segmentation.

Practical
1. Write program to read and display digital image using MATLAB or SCILAB
2. Become familiar with SCILAB/MATLAB Basic commands b. Read and display image in SCILAB/MATLAB

     a.   Resize given image
     b.   Convert given color image into gray-scale image
     c.   Convert given color/gray-scale image into black&white image f. Draw image profile
     d.   Separate color image in three R G & B planes
     e.   Create color image using R, G and B three separate planes
     f.   Flow control and LOOP in SCILA
     g.   Write given 2-D data in image file

3. To write and execute image processing programs using point processing method
     a.   Obtain Negative image
     b.   Obtain Flip image
     c.   Thresholding
     d.   Contrast stretching

4. To write and execute programs for image arithmetic operations
     a.   Addition of two images
     b.   Subtract one image from other image
     c.   Calculate mean value of image
     d.   Different Brightness by changing mean value

5. To write and execute programs for image logical operations
     a.   AND operation between two image
     b.   OR operation between two images
     c.   Calculate intersection of two images
     d.   Water Marking using EX-OR operation e. NOT operation (Negative image)
     e.   To write a program for histogram calculation and equalization using Standard MATLAB function

6. To write and execute program for geometric transformation of image
     a.   Translation
     b.   Scaling
     c.   Rotation
     d.   Shrinking
     e.   Zooming

7. To understand various image noise models and to write programs for
     a.   Image restoration
     b.   Remove Salt and Pepper Noise
     c.   Minimize Gaussian noise
     d.   Median filter and Weiner filter

8. Write and execute programs to remove noise using spatial filters
     a.   Understand 1-D and 2-D convolution process
     b.   Use3x3 Mask for low pass filter and high pass filter

9. Write and execute programs for image frequency domain filtering
     a.   Apply FFT on given image
     b.   Perform low pass and high pass filtering in frequency domain
     c.   Apply IFFT to reconstruct image

10. Write a program in C and MATLAB/SCILAB for edge detection using different edge detection mask.

11. Write and execute program for image morphological operations erosion and dilation.

12. To write and execute program for wavelet transform on given image and perform inverse wavelet transform to reconstruct image.

References

1. Gonzalez, R. C., &Woods, R. E. (2017).*Digital Image Processing*. 4th edition. Pearson Education.
2. Jain, A. K. (1988).*Fundamentals of Digital Image Processing*. 1st edition Prentice Hall of India.

Additional Resources
1. Castleman, K. R. (1995.). *Digital Image Processing*. 1st edition. Pearson Education
2. Gonzalez, R. C.,Woods, R. E., &Eddins, S. (2004).*Digital Image Processing using MATLAB*. Pearson Education Inc.
3. Schalkoff, D. (1989). *Image Processing and Computer Vision*. 1st edition. JohnWileyand Sons.

BCA822B: Data Mining
Credit: 06
Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)
Workload: 4 Lecturers (Per Week), 4 Practicals (Per Week)

Course objective
This course introduces data mining techniques and enables students to apply these techniques on real-life datasets. The course focuses on three main data mining techniques: Classification, Clustering and Association Rule Mining tasks.

Course Learning Outcomes
On successful completion of the course, students will be able to do following:
1. Pre-process the data, and perform cleaning and transformation.
2. Apply suitable classification algorithm to train the classifier and evaluate its performance.
3. Apply appropriate clustering algorithm to cluster data and evaluate clustering quality
4. Use association rule mining algorithms and generate frequent item-sets and association rules

Detailed Syllabus

Unit-I: 25 mark
Introduction to Data Mining-Applications of data mining, data mining tasks, motivation and
Challenges, types of data attributes and measurements, data quality.
Data Pre-processing-aggregation, sampling, dimensionality reduction, Feature Subset Selection, Feature Creation, Discretization and Binarization, Variable Transformation.

Unit-II: 25 mark
Classification: Basic Concepts, Decision Tree Classifier: Decision tree algorithm, attribute selection measures, Nearest Neighbor Classifier, Bayes Theorem and Naïve Bayes Classifier,
Model Evaluation: Holdout Method, Random Sub Sampling, Cross-Validation, evaluation metrics, confusion matrix.

Unit-**III: 25 mark**
Association rule mining: Transaction data-set, Frequent Items, Support measure, Apriori
Principle, Apriori Algorithm, Computational Complexity, Rule Generation, Confidence of
association rule.

Unit-IV: 25 mark
Cluster Analysis: Basic Concepts, Different Types of Clustering Methods, Different Types
of
Clusters, K-means: The Basic K-means Algorithm, Strengths and Weaknesses of K-means
algorithm, Agglomerative Hierarchical Clustering: Basic Algorithm, Proximity between
clusters, DBSCAN: The DBSCAN Algorithm, Strengths and Weaknesses.

Practical
Section 1: Preprocessing

Age	Age group	height	status	Years married
21	adult	6.0	single	-1
2	child	3	married	0
18	adult	5.7	married	20
125	elderly	5	widowed	2
34	child	-7	married	3

1. Create a file "people.txt" with the following data:
    a. Create a rule set *E* that contain rules to check for the following conditions:
    b. The age should be in the range 0-150.
    c. The age should be greater than years married.
    d. The status should be married or single or widowed.
    e. If age is less than 18 the age group should be child, if age is between 18 and
       65 the age group should be adult, if age is more than 65 the age group should
       be elderly.
    f. Check whether rule set *E* is violated by the data in the filepeople.txt.
    g. Summarize the results obtained in part C
    h. Visualize the results obtained in part C
2. Perform the following preprocessing tasks on the dirty_iris dataset
    a. Calculate the number and percentage of observations that are complete.
    b. Replace all the special values in data with NA.
    c. Determine how often each rule is broken (violated Edits). Also summarize and
       plot the result.
    d. Find outliers in sepal length using boxplot and boxplot.stats
3. Load the data from wine dataset. Check whether all attributes are standardized or not
   (mean is 0 and standard deviation is
    a. If not, standardize the attributes. Do the same with Iris dataset. Section 2: Data
       Mining Techniques
    b. Run following algorithms on 2 real datasets and use appropriate evaluation
       measures to compute correctness of obtained patterns:
4. . Run Apriori algorithm to find frequent item sets and association rules
    a. Use minimum support as 50% and minimum confidence as 75%
    b. Use minimum support as 60% and minimum confidence as 60 %

5. Use Naïve bayes, K-nearest, and Decision tree classification algorithms and build classifiers. Divide the data set into training and test set. Compare the accuracy of the different classifiers under the following situations:
   a. Training set=75%Test set = 25%
   b. Training set = 66.6% (2/3rd oftotal), Test set =33.3%
6. Use Simple K-means, DBScan, Hierachical clustering algorithms for clustering. Compare the performance of clusters by changing the parameters involved in the algorithms.

Recommended Datasets for Data Mining practicals
1. UCI Machine Learning repository.
2. KDD Datasets
3. Open data platform, Government of India(https://data.gov.in/)

References
1. Han, J., Kamber, M.,&Jian,P. (2011).*Data Mining: Concepts and Techniques*. 3rd edition. Morgan Kaufmann
2. Tan, P.-N., Steinbach,M., &Kumar,V. (2005).*Introduction to Data Mining*. 1st Edition. Pearson Education.

Additional Resources
1. Gupta, G. K. (2006).*Introduction to Data Mining with Case Studies*. Prentice-Hall of India.
2. Hand, D.,&Mannila,H. &Smyth, P. (2006).*Principles of Data Mining*. Prentice-Hall of India

BCA822C: Project Work/ Dissertation (Discipline Specific Elective)
Credit: 06
Total Marks: 100 Marks

Major Project

The Project is carried out by the student under guidance and support of faculty and management of the respective institute. It is felt that such a project provides an opportunity to the student to apply his/her knowledge and skills to real life problems (including oral and written communication skills), and as such the project should be given utmost importance and priority both by the students as well as institution faculty / management in respect of its identification, planning and implementation.

Objective of the Major Project

The aim of the project is to give the students an integrated experience in solving a real life problem by applying knowledge and skills gained on completion of theory papers in a course at a given Level. It provides an occasion for students to develop written and communication skills, Project also helps the students to realize the importance of resource and time management, ownership of task towards deliverables, innovation and efficiency in task

management apart from presentation skills. It also provides a good opportunity for students to build, enhance and sustain high levels of professional conduct and performance and evolves a problem solver frame of mind in student. It is also felt that taking up the project by a student prepares him for a job in industry and elsewhere.

Type of Major Project

Student can develop a project of any type like browser based, mobile based or client- server architecture base application. However, it should be based on specialized area which the student has opted. For example, if a student has opted Artificial Intelligence (AI), he/she has to develop a major project in the application areas of AI.

Project Group Organization/Plan
- Students will initially prepare a synopsis (500 words) and submit it to their respective department.
- For a given project, the group size could be a maximum of four (04) students.
- Each group will be assigned a teacher as a supervisor
- A maximum of four (04) projects would be assigned to one teacher.

Some Important Notes While Preparing the Project Proposal

The following suggested guidelines may be followed in preparing the Final Project Report: Good quality white executive bond paper A4 size should be used for typing and duplication. Care should be taken to avoid smudging while duplicating the copies. Page Specification: (Written paper and source code)
- ➢ Left margin 3.0 cms Right margin 3.0 cms
- ➢ Top margin 2.7 cms
- ➢ Bottom margin 2.7 cms
- ➢ Page numbers -All text pages as well as Program source code listing should be numbered at the bottom center of the pages

Submission of Project Report

The student will submit his/her project report in the prescribed format. The Project Report should include:
- ➢ One hard copy of the Project Report.
- ➢ Soft copy of Project on Optical Drive
- ➢ The Project Report may be about 50 pages (excluding coding).

Project Evaluation

100 marks for end semester examination comprising Viva/presentation (50 marks) and project report evaluation (50 marks): to be awarded jointly by the examiner and supervisor/ mentor.

Skill-Enhancement Elective Course (SEC-I)

BCA123A: Mathematical Foundation (Skill-Enhancement Elective Course)
Credit: 06
Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)
Workload: 4 Lecturers (Per Week), 4 Practical (Per Week)

Course Objectives:
The objective of the course is to introduction the Mathematical Foundation which Provide a strong foundation in fundamentals of computers. Prepare the students with exceptional skills of problem solving, communication and leadership skills. Facilitate overall understanding of the requirements of the subjects. Prepare the students to provide professional solutions to real time problems.

Course Outcomes:
On completion of the course students will be able to
1. Study and solve problems related to connectives, predicates and quantifiers under different situations.
2. Develop basic knowledge of matrices and to solve equations using Cramer's rule.
3. Know the concept of Eigenvalues.
4. To develop the knowledge about derivatives and know various applications of differentiation.
5. Understand the basic concepts of Mathematical reasoning, set and functions

Detailed Syllabus

UNIT-I: 15 mark
Algebra: Logarithms- Introduction, Definition, Laws of operations, change of base, Binomial theorems-Introduction, Binomial theorem, Position of terms. Analytical

geometry: Introduction, directed line, midpoint, distance between two points, Section formula, external division, coordinates of a centroid, Area of a triangle. The straight line–slope of a straight-line, different forms of equations of the straight-line.
Circle-The equation of a circle, different forms of circles, General Equation of the circle.

UNIT-II: 15 mark
Matrix Algebra: Definition, types of matrices, algebra of matrices–addition of matrices, subtraction of matrices, multiplication of matrices, determinant of a matrix, Adjoin of a matrix, orthogonal and unitary matrix, rank of a matrix, echelon form of a matrix, normal form of a matrix, equivalence of matrices

UNIT-III: 15 mark
Inverse of a matrix, Characteristic equation of a matrix,Cayley Hamilton theorem, Eigenvalues. System of Linear equations: solution of linear homogeneous and non-homogeneous equations(matrix method), Cramer's rule Arithmetic progression: Definition, formula term, sum of terms, Arithmetic mean, problems
Geometric

UNIT-IV: 15 mark
Trigonometry: Introduction, Measurement of angles, trigonometric functions, relation between trigonometric functions, signs of trigonometric functions, trigonometric functions of standard angles.

UNIT-V: 15 mark
Calculus: Limit of function, continuity of a function. Differentiation: Derivative of a function one variable, Power function, constant with a function, sum of functions, product of two functions, quotient of two functions.
Integration-Indefinite integral, rules of integration, some standard results and examples, definite integral

Text book:
1. C Sanchethi and VK Kapoor, Business Mathematics, Sulthan Chand&Sons Educational publishers, New Delhi, Eleventh Revised Edition.
2. P.R.Vittal-Business Mathematics and Statistics, Margham Publications,Chennai
3. Pundir &S.K.Pundir, A Text Book of BCA Mathematcis-I, Rimple A, Pragatis Edition
4. B.S.Vatsa-Discrete Mathematics–New Age International Limited Publishers, New Delhi

BCA123B: Office Automation Tools
Credit: 06
Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)
Workload: 4 Lecturers (Per Week), 4 Practicals (Per Week)

Objective of the Course:
To provide an in-depth training in use of data entry, internet and internet tools. The course also helps the candidates to get acquainted with IT.

Learning Outcomes:
After completion of the course, students would be able to documents, spreadsheets, make small
presentations and would be acquainted with internet and will have good typing speed.

Detailed Syllabus
Unit I: 15 mark
Introduction to computer: History, characteristics, components. Concepts of Hardware and Software Representation of Data/Information. Concept of Data processing. Applications of IECT: e-governance & Multimedia and Entertainment.

Unit II: 15 mark
Introduction to GUI Based Operating System. Basics of WINDOWS. The User Interface. File and Directory Management. Elements of word processing. Text Creation and manipulation. Text Formatting- Paragraph, Indenting. Bullets and Numbering, Changing case. Macros. Table Manipulation: Draw Table, Changing cell width and height, Alignment of Text in cell, Delete / Insertion of row and column, Border and shading Cross reference, table of contents. Mail Merge.

Unit   III: 15   mark
15M
Spread sheet elements. Manipulation of Cells: Entering Text, Numbers and Dates , Creating Text, Number and Date Series, Editing Worksheet Data, Inserting and Deleting Rows, Column, Changing Cell Height and Width
Using Formulas. Working with function and charts. Advanced filter, comments, protecting worksheet

Unit   IV: 15   mark
15M
Power point presentations: Creation of Presentation, Preparation of Slides. Transition and Slide Timing and Automating a Slide Show.
Basics of Computer Networks. Services on Internet. ISPs and examples (Broadband/Dialup/WiFi). TCP/IP and OSI layers. WWW, web browser and Search Engines.

Unit   V: :   15   mark
15M
Basics of E-mail. Opening Email Client. Mailbox: Inbox and Outbox. Instant Messaging and Collaboration.
Soft Skills: Personality Development. Writing Skills, File Handling. Data Entry / Typing computer.

Practical List (use Microsoft office or Libre office)

PRACTICAL EXERCISE 1.1

Using the data given, get the sum of all the figures within the range.

	A	B	C	D	E	F	G
1		Mon	Tue	Wed	Thur	Fri	TOTAL
2	Breakfast	3,560	3,186	2,952	3,395	3,436	
3	Lunch	20,163	21,416	19,912	19,681	18,628	
4	Bar	9,873	12,172	12,642	12,711	18,846	
5	Snacks	2,405	3,544	2,694	3,120	3,712	
6	TOTALS						

PRACTICAL EXERCISE 1.2

Enter the data given below into a worksheet.

	A	B	C	D	E
1	**Stationery Supplies Ltd**				
2					
3	**Date**	**SalesPerson**	**Item**	**Receipt No**	**Amount**
4	21-Nov	Carl	Toys	1238	1,782.10
5	26-Nov	Carl	Stationery	1255	4,853.55
6	26-Nov	Carl	Toys	1395	51.35
7				**Carl's Total**	
8	21-Nov	John	Cards	1141	91.15
9	24-Nov	John	Books	1982	442.60
10	21-Nov	John	Toys	1885	561.50
11	26-Nov	John	Toys	1875	62.75
12				**John's Total**	
13	22-Nov	Judy	Books	1032	234.50
14	26-Nov	Judy	Sports goods	1920	472.60
15				**Judy's Total**	
16	25-Nov	Mary	Toys	1774	364.15
17				**Mary's Total**	
18	22-Nov	Susan	Electronics	1160	52.95
19	23-Nov	Susan	Cards	1075	81.60
20	23-Nov	Susan	Others	1745	132.95
21	24-Nov	Susan	Sports goods	1662	2,580.10
22				**Susan's Total**	
23					
24				**Grand Total**	

Calculate the totals for each salesperson and get the grand total.:

Format the worksheet as follows:

Make all the Totals bold, two decimal places, comma, center the title across columns A-E and make it size 16, bold and Italic.

Put a double border round the whole table and a single line border inside the table.

Save the worksheet as **Stationery Analysis**.

PRACTICAL EXERCISE 1.3

A Payroll consists of Basic Pay, Allowances, Gross Salary, Deductions and Net Salary. The Allowances are 23% of the Basic Pay while the Deductions are 12% of the Gross Salary.

In the given worksheet, indicate in each cell what will be inserted, that is – a value or a formula. In the case of a formula, write down the formula in the cell.

	A	B	C	D	E	F
1	Stationery Supplies Ltd					
2						
3	Name	Basic Pay	Allowances	Gross Salary	Deductions	Net Salary
4	Lewis					
5	Francis					
6	Edwin					
.						.
.						.
.						.
13	Totals					

PRACTICAL EXERCISE 1.4

Assume you are the Accountant of Stationery Supplies Ltd. Below is the current payroll in the workbook OLD PAYROLL.

OLD PAYROLL.

	A	B	C	D	E	F
1	Stationery Supplies Ltd.					
2						
3	Name	Basic Pay	Allowances	Gross Salary	Deductions	Net Salary
4	Lewis	15,791	3,137	18,928	1,256	17,672
5	Francis	15,537	3,061	18,598	776	17,822
6	Edwin	15,506	3,051	18,557	999	17,558
7	Bernard	15,417	3,025	18,442	1,099	17,343
8	George	15,008	2,902	17,910	718	17,192
9	Albert	14,969	2,890	17,859	846	17,013
10	Edward	14,651	2,795	17,446	760	16,686
11	Cornell	14,618	2,785	17,403	663	16,740
12	John	14,553	2,765	17,318	558	16,760
13	Carl	14,508	2,752	17,260	706	16,554
14						
15	Totals	150,558	29,163	179,721	8,381	171,340

The following salary review is given to you in the workbook INCREMENT.

INCREMENT

	A	B	C
1	Name	Current Pay	% increase
2	Lewis	15,791	19%

3	Francis	15,537	19%
4	Edwin	15,506	22%
5	Bernard	15,417	18%
6	George	15,008	21%
7	Albert	14,969	17%
8	Edward	14,651	15%
9	Cornell	14,618	25%
10	John	14,553	19%
11	Carl	14,508	20%
12			
13	**Allowances**		
14	**20%**		

Required:

Using formulas, you are required to update the payroll with the changes in a blank worksheet. This new blank sheet is in the workbook NEW PAYROLL.

PRACTICAL EXERCISE 1.5

You are in charge of a young and growing business. You have identified the various factors (sources of revenue and expenses) that influence the business as shown in the table below. Use the figures provided and the layout to create a financial projection model for the business for the next six years. The parameters are given on Sheet 2.

SHEET 1:

INCOME AND EXPENSES PROJECTIONS						
	1999	**2000**	**2001**	**2002**	**2003**	**2004**
Sales	10,000					
% Growth over the previous year		20%	30%	20%	10%	10%
Materials						
Wages						
Other benefits						
Others						
Total Cost of Goods Sold						
Salary: Office						
Salary: Sales						
Other Benefits						
Advertising & Promotions						
Depreciation						
Miscellaneous						
Total General & Admin. Expenses						
Total Operating Costs						

Interest on Loans						
Pre-tax Income						
Tax						
Profit						

SHEET 2:

Parameters		Description
Sales	10,000	Starts at 10,000 and grows by a percentage
Materials	17%	17% of Sales
Wages	14%	14% of Sales
Other benefits	2.1%	2.1% of Sales
Others	8%	Starts at 100, then grows by 8% yearly
Salary: Office	10%	Starts at 1,000, then grows by 10% annually
Salary: Sales	8%	8% of Sales
Other Benefits	17%	17% of Total Salary
Advertising & Promotions	2.5%	2.5% of Sales
Depreciation	20	Fixed at 20 every year
Miscellaneous	10	Starts at 10 and grows by a fixed amount of 10 annually
Interest on Loans	10	A fixed amount of 10 each year
Tax	52%	52% of Pre-tax Income

Exercise Instructions.

Open the worksheet named **Income and Expenses Projections.xls**.
Rename Sheet1 as **Projections** while Sheet 2 should now be **Parameters**.
Calculate the Sales for the year 2000 using the percentage given in cell **C5**.
Copy the formula across to the Year 2004.
Calculate the different items that make up the Total Operating Costs using the parameters in the Parameters sheet.
(*You should enter the formula for the Year 1999 and copy down to the year 2004. Use Absolute Referencing effectively*).
Hint: *Total Cost of Goods Sold = Materials + Wages + Other Benefits + Others*

Calculate the Total Operating Costs:

Total Cost of Goods Sold + Total General and Administrative Expenses.

Calculate the Interest on Loans:
Calculate the Pre-tax Income.

Sales – Total Operating Cost – Interest on Loans.

Calculate the Tax.
Calculate the Profit:

Pre-tax Income - Tax.

Format the worksheet as follows:
Make all the Totals bold, zero decimal places, comma, center the heading between A1:G1 and make it size 16, bold.
Save the file as **C:\Exams\Creative.xls**

PRACTICAL EXERCISE 1.6

From the data given in the table below, create a Pie Chart to show the distribution of the total amount amongst the various salesmen.

	A	B	C	D	E	F
1	**ABC Company Sales Performance Report**					
2						
3	**Salesman**	**Qtr1**	**Qtr2**	**Qtr3**	**Qtr4**	**Total**
4	Albert	148	156	171	140	615
5	Carl	122	131	153	118	524
6	Cornell	211	243	246	250	950
7	Edwin	129	150	92	218	589
8	Francis	311	270	247	322	1,150

PRACTICAL EXERCISE 1.7

	A	B	C	D	E	F	G	H	I
1	**XYZ COMPANY SALES PERFORMANCE**								
2									
3									
4	**SALESPERSON**	**ANNUAL TARGET**	**QTR1**	**QTR 2**	**QTR 3**	**QTR 4**	**TOTAL SALES**	**AVERAGE SALES**	**COMMISSION**
5									
6	ALBERT	750	148	256	133	154	X		
7	MICHAEL	650	187	143	258	143	X		
8	CARL	800	233	200	216	152	X		
9	GEORGE	700	256	145	136	259	X		
10	LUCY	1,000	249	212	215	124	X		
11									
12	TOTAL	X	X	X	X	X	X		
13									
14	COMMISSION	6%							

15											

Given the table above, write formulas or describe how you would calculate the Total Sales in column G and Row 12.

How would you calculate the Average Sales? Write the formula as it should appear in Excel and show the method of duplicating it to the other cells.

How would you calculate the Commission? Write the formula as it would appear in Excel and explain the method of duplicating it to the other cells.

Explain how you can insert two rows above Row 10.

 What does this sign (# # # # # # #) mean when seen in cell(s)? What should you do when you see this sign?

How does one delete an entire row or column and all its cells?

You are asked to compare QTR1 and QTR3 sales for all salespersons in the above table using a chart:

What range of cells do you need to select and how would you select it?

What type of graph would you use?

How can you edit a chart once it is created, lets say, to change the series from columns to rows, to change the legend, etc?

PRACTICAL EXERCISE 1.8

Use the Balance Sheet given below to answer the questions that follow:

	1999	2000	2001	2002	2003	2004
Income and Expenses Projection						
	1999	**2000**	**2001**	**2002**	**2003**	**2004**
Sales	10,000	12,000	15,600	18,720	20,592	22,651
% Growth over the previous year		20%	30%	20%	10%	10%
Materials	1,700	2,040	2,652	3,182	3,501	3,851
Wages	1,400	1,680	2,184	2,621	2,883	3,171
Other benefits	210	252	328	393	432	476
Others	100	108	117	126	136	147
Total Cost of Goods Sold	3,410	4,080	5,280	6,322	6,952	7,644

Questions

Insert five rows above Row 1.
Reduce the size of the picture to a **Height of 0.96"** by a **Width of 1.66" (inches).**
Drag and drop the picture to fit in the range A1:A5.
Add the Header '**Balance Sheet 2001**' to the right section of the worksheet.
Make the heading size 14, and Bold.
Save the file as **Balance Sheet**.

PRACTICAL EXERCISE 1.9

Use the worksheet given below to answer the questions that follow:

Employee Details						
Emp No	Name	Category	Basic Pay	Allowances	Gross Pay	Tax Deductions
E8	Cornell	Assistant	4,600	300		
E9	John	Assistant	3,500	450		
E2	Francis	Supervisor	6,508	500		
E3	Edwin	Management	8,006	1,801		
E10	Carl	Assistant	4,200	100		
E4	Bernard	Management	7,917	1,775		
E5	George	Supervisor	5,500	850		
E1	Lewis	Management	8,291	2,500		
E6	Albert	Supervisor	5,700	760		
E7	Edward	Supervisor	7,151	1,545		
Tax Rate						
12%						

Questions

Make all the column titles bold, and size 12. Center the title , across columns A1:H1 and make it size 16, and Bold.
Calculate the gross pay in cell **F4**.
Calculate the amount of tax deducted from each employee, given that the tax rate is 12% of the gross pay. Tax rate is found in cell A16 of the worksheet.
Format the text orientation in the range **A4:G4** to 0 degrees.
Adjust the column width such that all the headings are visible.
In cell **H4**, enter the title **Net Pay** and calculate the Net pay for all employees.
Format the range **G4:H13** to **zero** (0) decimal places.
Format the title **Net Pay** to match the other titles.
Set the range A1 to H16 as **Print Area**.
Using the Names in column B, Basic Pay in column D, and Allowances in column E, insert a **Clustered Column Pie Chart** on the same sheet to show comparison of the salaries for the

employees. The Chart Title should be **Employee Details**, the Y-axis should be **Employee Names** and the X-axis should be **Thousands (Kshs).**

Move the chart so that the top left corner is on cell A18.

Change the Chart Title to **Employees' Salary Details**.

Change the Chart Type to **Clustered bar with a 3-D visual effect**.

Increase the Chart Title **Font size to 14**.

Change the text direction for the title of the X-axis to 0 degrees, and for the Y-axis to -90 degrees.

Resize the chart such that the bottom left corner is on cell A55, while the bottom right corner is on cell I55 so that all the details are clearly visible.

Save the worksheet as **Salary Details (Reviewed Copy)**.

PRACTICAL EXERCISE 1.10

You have been asked to analyse the rainfall pattern of a city called Mooncity. The following data, which shows the average weekly and monthly rainfall, has been provided.

Table 1.

Weekly Rainfall of Mooncity in mm.

MON	TUE	WED	THUR	FRI	SAT	SUN
10	5	30	20	15	0	50

Table 2.

Monthly Rainfall of Mooncity in mm.

JAN	FEB	MAR	APR	MAY	JUN	JUL	AUG	SEP	OCT	NOV	DEC
30	25	55	100	60	40	35	60	35	25	20	10

Enter Table 1 and Table 2 into Microsoft Excel and Save as **Mooncity**

Using the most appropriate formulas and functions,

Calculate the total rainfall for the week and the year respectively.

Find the lowest rainfall for the week and the year respectively.

Find the highest rainfall for the week and the year respectively.

Find the mean rainfall for the week and the year respectively.   .

Save the changes made to the worksheet.

Create a 3-D column chart for Table 1. The title for the graph should be **AVERAGE DAILY RAINFALL**. The title for the X-axis should be **DAY** while that of the Z-axis should be labeled as **RAINFALL (mm).** The chart should be inserted as an object within the worksheet. Format the chart appropriately

Practical List for PowerPoint:
1. Create five Power point slides. Each slide should support different format. In these slides explain areas of applications of IT. Make slide transition time as 10 seconds.
2. Create five Power Point slides to give advantages/disadvantages of computer application of computers and logical structure of computer.
3. Create five Power Point slides detailing the process of internal assessment. It should be a self-running demo.

Books Recommended:
1. O level ccc bpb publication

2. Sushila Madan , Introduction to Essential tools,JBA,2009.
3. Anita Goel, Computer Fundamentals, Pearson,2012

BCA123C: PC Hardware and Maintenance (BC123C) SEC-2
Credit: 06
Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)
Workload: 4 Lecturers (Per Week), 4 Practicals (Per Week)

Learning Objective:
The objective of the course is to acquire basic knowledge in Computer hardware and peripherals for installation, trouble shooting and maintenance including system software management and its back up and to undertake disaster prevention, diagnosis and rectification of faults besides personality development and communication skills.

Learning Outcomes:
Students will get opportunities to get jobs in the sectors like Service Industry (Hardware and Software) and Academia as Troubleshooters, Technicians (PC), Assistant System Administrators, and Lab Demonstrators.

Detailed Syllabus

UNIT-I: 15 mark
Identify and handling of Internal components in the PC Cabinets like SMPS and its connection to Motherboard and various devices, Level 4 Identify Motherboard, CPU, Chipset, Slots, Memory modules, memory slots, Hard Disc Drives, CDROM/DVD/Blue-Ray Disc, etc.

UNIT-II: 15 mark
Assembly of PC using various parts, Interconnection between devices, cable polarities and connections, SMPS installation and power connection. Identify and putting various types of Add-on Cards, Motherboard slot and their application

UNIT-III: 15 mark
Testing & troubleshooting of personal computer problems interfacing primary Memory card in the CPU. Interfacing Hard disk with the motherboard. Testing &Troubleshooting of PC through various tests. Installation of multiple HDD and creation of single large volume out of it,

UNIT-IV: 15 mark
Installation of Operating System like Windows7, Windows8, Windows 2008, and various Linux flavors like Ubantu, SUSE, Red Hat, and Introduction to VMware Virtualization etc. Troubleshooting of various hardware problems like SMPS failure, Display not there, missing OS or re-installation of user software or system software.

UNIT-V: 10 mark

CDROM, DVD lens cleaning or replacement, CMOS setup, Battery replacement on motherboard in case BIOS is not retaining correct values.

UNIT-VI: 10 mark

Driver software downloading and installation, Antivirus Software installation, scanning for viruses, removing .tmp files from WINDOWS machine, etc

Practical:
1. Identification of the peripherals of a computer, components in a CPU and their functions.
2. Assembling and disassembling the system hardware components of personal computer.
3. Basic Computer Hardware Trouble shooting.
4. LAN and WiFi Basics.
5. Operating System Installation – Windows OS, UNIX/LINUX, Dual Booting.
6. Installation and Uninstallation of Software – Office Tools, Utility Software (like Anti-Virus, System Maintenance tools); Application Software - Like Photo/Image Editors, Audio Recorders/Editors, Video Editors ...); Freeware, Shareware, Payware and Trialware; Internet Browsers, Programming IDEs,
7. System Configuration – BIOS Settings, Registry Editor, MS Config, Task Manager, System Maintenance, Third-party System Maintenance Tools
8. Troubleshooting hardware.

Reference
1. Adane Nega Tarekegn, Alemu Kumilachew Tegegne, "A simple guide to Computer Maintenance and Troubleshooting", Lambart Publication.
2. Craig Zacker, John Rourke,"Pc Hardware, A complete reference ",McGraw Hill Manahar Lotia, "Modern Computer Hardware Course", BPB Publication

Skill-Enhancement Elective Course (SEC-2)

BCA224A: Linux Environment (Skill-Enhancement Elective Course)
Credit:04
Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)
Workload: 4 Lecturers (Per Week), 4 Practicals (Per Week)

Course Objectives
1. To understand and make effective use of linux utilities and shell scripting language to solve problems
2. To implement in C some standard linux utilities like mv,cp,ls etc…
3. To Develop the skills the necessary for systems programming including file system programming, process and signal management and inter process communication
4. To develop the basic skills required to write network programs using sockets

Course Outcome:
On completion of course students will be able to:

1. Be familiar with fundamentals of Linux operating system.
2. To learn the concepts of files and file organization.
3. To learn the mechanisms involved in ownership of files and file attributes
4. To gain the knowledge on vi editor and regular expressions
5. To know the techniques of shell programming.

Detailed Syllabus

UNIT-I: 25 mark
Introduction to Unix: Brief History, What is Unix?, Unix Components, Using Unix, Commands in Unix, Some Basic Commands, Getting Help, Command Substitution, Giving Multiple Commands, Aliases.
Files and File Organization: Unix Files, Categories of Files, Hidden Files, File System, Path Names, Home Directory, Directory Commands, File Related Commands, Wild Cards, Displaying the Contents of a File, Printing of Files, Comparing Files.

UNIT-II: 25 mark
File Attributes and Permissions: Ownership of Files, File Attributes, File Command, Changing File Permission, Changing the Owner of a File, Changing the group of a File, Times Associated with a File, umask Command.
Standard I/O, Redirection Pipes and Filters: Standard I/O, Redirection, Pipe and Pipeline, Filter, tee command, Terminal files, Sample Database, Handling Columns and Fields, sort command, uniq command, tr command.

UNIT-III: 15 mark
The vi Editor: vi Editor, Editing with vi, Moving the Cursor, Editing, Copying and Moving Text, Pattern Searching, Repeating the Last Editor Command, Undoing Commands, Joining Lines, Writing Selected Lines onto a Separate File, Using the Shell from vi, Configuring the vi Environment.**Regular Expressions:** grep Family of Commands and sed : Regular Expressions, grep Family, egrep Command, fgrep Command, Stream Editor-sed.

UNIT-IV: 25 mark
Shell Programming : Shell Variables, export Command, .profile File – A Script Run during Starting, The First Shell Script, read Command, Positional Parameters, The $? Variable – Knowing the Exit Status, More about the set Command, exit Command, Branching Control Structures, Loop-Control Structures, continue and break Statements, expr Command, Real Arithmetic in Shell Programs, The here Document (<<), sleepCommand, Debugging Scripts, script Command, eval Command, exec Command

Text Books
1. M.G.Venkateshmurthy: Introduction to Unix & Shell Programming, Pearson Education.

References
2. John Goerzen: Linux Programming Bible, IDG Books, NewDelhi.
3. Sumitabha Das: Your Unix - The Ultimate Guide,TMH.
4. Richard Petersen: The Complete Reference – Linux,McGraw-Hill
5. Yashwant Kanetkar: Unix & Shell programming –BPB

Practical:

1. Write a program to print the current date, month and year; and display both the date and time in the format dd/mm/yy/hh:mm:ss.
2. Write a program to create five directory, list all the directory using the wild card * and remove the directory iteratively
3. Write a program to create a file, count the number of lines, words and characters of that file
4. Write a program to create two files and display only the lines common to two files.
5. Write a program to
   a. Display all the users along with the column headers
   b. Identify directories and executable in a list
   c. Show all hidden files.
6. Write a program to calculate the sum of five digit number.
7. Write a program to calculate the area, perimeter of the rectangle, and the area & circumference of the circle. [Length and breadth of a rectangle and radius of a circle are input through the keyboard]
8. Write a shell script to calculate the profit and lost.
9. Write a program to calculate gross salary if the DA is 45% of basic salary, and House Rent Allowance is 20% of basic salary.
10. Write a program to find out whether an input integer is an odd number or even number.
11. Write a program to determine whether the year is leap or not. [Use the logical operators –a and –o]
12. Write a shell script which displays information about a given file in proper format.
13. Write a shell script which receives two file names as an argument, checking whether the two file's contents are same or not. If same, the second file should be deleted.
14. Write a program to calculate overtime pay of 10 employees. Overtime is paid at the rate of Rs 15.00 per hour for every hour worked above 35 hours.
15. Write a program to fine the factorial value of a given number.
16. Write a shell script which deletes all lines containing the word *'programming'* in the files supplied as an arguments to this shell script.
17. Write a shell script which displays a list of all files in the current directory to which you have read, write, and execute permission.
18. Write a shell script to check whether the given argument is file or a directory and display the number of lines if it is a file.
19. Write a shell script to print prime number from 1 to 100.

<div align="center">

BCA224B: Object Oriented Programming Using C++() Sec-2

Credit: 06

Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)
Workload: 4 Lectures (Per Week), 4 Practicals (Per Week)

</div>

Course Objective:

1. To learn the fundamental programming concepts and methodologies which are essential to building good C/C++ programs.
2. To practice the fundamental programming methodologies in the C/C++ programming language via laboratory experiences. Microsoft Visual Studio is the programming environment that will used.
3. To code, document, test, and implement a well-structured, robust computer program using the C/C++ programming language.
4. To write reusable modules (collections of functions).

Learning Outcomes:

1. Demonstrate an understanding of algorithms in the problem-solving process.
2. Identify the necessary properties of good problem-solving techniques.
3. Create and analyze algorithms for solving simple problems.
4. Use incremental program development to create, test, and debug algorithms for solving simple problems.

UNIT-I: 10 mark

Introduction to C++: Difference between C and C++- Evolution of C++- The Object-Oriented Technology- Disadvantage of Conventional Programming- Key Concepts of Object-Oriented Programming Advantage of OOP- Object-Oriented Language.

UNIT-II: 15 mark

Classes and Objects &Constructors and Destructor: Classes in C++-Declaring Objects-Access Specifiers and their Scope- Defining Member Function-Overloading Member Function- Nested class, Constructors and Destructors, Introduction- Constructors and Destructor- Characteristics of Constructor and Destructor- Application with Constructor-Constructor with Arguments (parameterized Constructor-Destructors- Anonymous Objects.

UNIT-III: 15 mark

Operator Overloading and Type Conversion & Inheritance: The Keyword Operator-Overloading Unary Operator- Operator Return Type- Overloading Assignment Operator (=)-Rules for Overloading Operators, Inheritance, Reusability- Types of Inheritance- Virtual Base Classes- Object as a Class Member- Abstract Classes- Advantages of Inheritance-Disadvantages of Inheritance,

UNIT-IV: 15 mark

Pointers & Binding Polymorphisms and Virtual Functions: Pointer, Features of Pointers-Pointer Declaration- Pointer to Class- Pointer Object- The this Pointer- Pointer to Derived Classes and Base Class, Binding Polymorphisms, and Virtual Functions, Introduction-Binding in C++- Virtual Functions- Rules for Virtual Function- Virtual Destructor.

UNIT-V: 10 mark

Generic Programming with Templates & Exception Handling: Generic Programming with Templates, Need for Templates- Definition of class Templates- Normal Function Templates- over Loading of Template Function-Bubble Sort Using Function Templates-Difference between Templates and Macros- Linked Lists with Templates, Exception Handling- Principles of Exception Handling- The Keywords try to throw and catch- Multiple Catch Statements –Specifying Exceptions.

Unit-VI: 10 mark

FileI/O : Opening and closing a file (use offstream header file, ifstream, ofstream and fstream classes), Reading and writing Text Files, Using put(),get(),read() and write() functions, Random access in files.

Text Book:
1. Object Oriented Programming with C++ By REEMA THAREJA
2. E Balaguruswamy, "Object Oriented Programming with C++", Tata McGraw-Hill Education, 2008.
3. Object Oriented Programming in TURBO C++ ,Galgotia, Robert Lafore
4. Computer Science with C++, Sumita Arora, Dhanpat Rai & Co.

References:
1. HerbtzSchildt, "C++: The Complete Reference", Fourth Edition, McGraw Hill.
2. BjarneStroustrup, "The C++ Programming Language", 4th Edition, Addison-Wesley , 2013.
3. BjarneStroustroup, "Programming -- Principles and Practice using C++", 2nd Edition, Addison-Wesley 2014.
4. Paul Deitel, Harvey Deitel, "C++ How to Program", 8th Edition, Prentice Hall, 2011.
5. John R. Hubbard, "Programming with C++", Schaum's Series, 2nd Edition, 2000.

Object oriented Programming using C++Lab

1. WAP to print the sum and product of digits of an integer
2. WAP to reverse a number.
3. WAP to compute the sum of the first n terms of the following series $S=1+1/2+1/3+1/4+\ldots\ldots$
4. WAP to compute the sum of the first n terms of the following series $S=1-2+3-4+5\ldots\ldots\ldots$
5. Write a function that checks whether a given string is Palindrome or not. Use this function to find whether the string entered by user is Palindrome or not.
6. Write a function to find whether a given no. is prime or not. Use the same to generatThe prime numbers less than 100.
7. WAP to compute the factors of a given number.
8. Write a macro that swaps two numbers. WAP to use it.
9. Write a program which takes the radius of a circle as input from the user, passes it to another function that computes the area and the circumference of the circle and displays the value of area and circumference from the main() function.
10. .Write a program to find sum of n elements entered by the user.To write this program, allocate memory dynamically using malloc()/ calloc() functions or new operator.
11. Given two ordered arrays of integers, write a program to merge the two-arrays to get an ordered array.
12. Create the Person class. Create some objects of this class (by taking information from the user). Inherit the class Person to create two classes Teacher and Student class. Maintain the respective information in the classes and create, display and delete objects of these two classes (Use Runtime Polymorphism).
13. Create a class triangle. Include overloaded functions for calculating area. Overload assignment operator and equality operator.
14. Create a class Box containing length, breath and height. Include following methods in it

        a. Calculate surface Area
        b. Calculate Volume

15. Write a program which takes input from the user for length, breath and height to test the above class.
16. Create a structure Student containing fields for Roll No., Name, Class, Year and Total Marks. Create 10 students and store them in a file.
17. Write a program to retrieve the student information from file created in previous question and print it in following format: Roll No.   Name Marks
18. Copy the contents of one text file to another file, after removing all whitespaces.
19. Write a function that reverses the elements of an array in place. The function must accept only one pointer value and return void.
20. Write a program that will read 10 integers from user and store them in an array. Implement array using pointers. The program will print the array elements in ascending and descending order.
21. Implement some inheritance program.
22. Exception handling program
23. Function template program

BCA224C: Digital Electronics SEC-2
Credit: 06
Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)
Workload: 5 Lecturers (Per Week), 1 Tutorial (Per Week)

Objective
1. To design digital circuits using simplified Boolean functions
2. To analyze and design combinational circuits
3. To analyze and design synchronous and asynchronous sequential circuits
4. To understand Programmable Logic Devices.

Outcome
On Completion of the course, the students should be able to:
1. Simplify Boolean functions using K Map
2. Design and Analyze Combinational and Sequential Circuits
3. Implement designs using Programmable Logic Devices
4. Write HDL code for combinational and Sequential Circuits

Unit-I:15 mark
Signal: Analog and discrete time signals, Digital signal, Importance of digital signal, Switch and bits, **Number Systems** Different number systems**,** Number system conversion**,** Arithmetic operations**,** r's and (r-1)'s complement**, Codes** Excess-3 code**,** Gray code**,** 2421 code**,** Binary coded decimal**,** Code **Logic Gates** Basic logic, Universal logic gates, Exclusive logic gates, Properties of logic gates **Boolean algebra:** SOP and POS forms, Complement and Dual forms.

Unit-II: 15 mark
Gate Level Minimization Karnaugh-Map introduction**,** 2,3 and 4 variable K-map', 5 variable K-map**,** K-map with don't care terms**,** Quine-McCluskey minimization technique, **Combinational Logic** Combinational circuits introduction**,** Half & full adder, Half & full

subtractors**,** CLA & BCD adder, Multiplexers & Demultiplexers**,** Encoders & Decoders**,** Seven segment display decoder**,** Code conversion.

Unit-III: 15 mark
Flip Flops Sequential circuits introduction NOR & NAND latch, Triggering methods, SR, D, JK & T flip flops, Flip flop conversion, Master slave flip flop, State table, state digaram & state equation, Mealy & Moore state machine, Analysis of clocked sequential circuits.

Unit-IV: 15 mark
Counters Introduction to counters, Asynchronous up counters, Asynchronous down counters, Asynchronous up/down counter, Synchronous up counters, Synchronous down counters, Synchronous up/down counter, Decade (BCD) counter, Johnson's counter & Ring counter .

Unit-V: 15 mark
Registers: Introduction to registers, Data formats & classification of registers, Shift registers (SISO, SIPO, PISO,PIPO), Bidirectional shift register, Universal shift register.
Memory Devices **-** RAM, ROM, PAL & PLA.

TEXT BOOK:
1. M. Morris R. Mano, Michael D. Ciletti, Digital Design With an Introduction to the Verilog HDL, VHDL, and System Verilog, 6th Edition, Pearson Education, 2017.

REFERENCES:
2. K. Kharate, Digital Electronics, Oxford University Press, 2010 John F. Wakerly, Digital Design Principles and Practices, Fifth Edition, Pearson Education, 2017.
3. Charles H. Roth Jr, Larry L. Kinney, Fundamentals of Logic Design, Sixth Edition, CENGAGE Learning, 2013
4. Donald D. Givone, Digital Principles and Design‖, Tata Mc Graw Hill, 2003.

Other Discipline:
Credit: 06 each
GEC-1 to GEC-6

BCA101: Fundamentals of Information Technology (General Elective Course)
Credit: 06
Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)
Workload: 4 Lecturers (Per Week), 4 Practicals (Per Week)

Course Objective
This course is designed to help the students to learn the basic study of computer hardware, operating systems, networking, Internet, databases, etc.

Course Learning Outcomes
On successful completion of the course, students will be able to:
1. Understand the concept of hardware and software.
2. Acquainting with input and output devices.

3. Understand networking concepts and models.
4. Learn and aware of Internet activities.

Detailed Syllabus

UNIT-I: 15 mark
What is Computer? Representation of data/information. What is Data Processing? Characteristics of a Computer System, Evolution of Computer, Generation of computers, Block Diagram of Digital Computer, Classification of Digital Computers Classification of Computers, Functional Components of a computer- Central Processing Unit, Memory- Primary Memory-RAM, ROM, Booting, Secondary Storages Devices : Floppy and Hard Disks, Optical Disks CD-ROM, DVD, Mass Storage Devices : USB thumb drive. Input /Output Devices- Keyboard, Mouse, Trackball, Joystick, Digitizing Tablet, Scanners, Digital Camera, MICR, OCR, OMR, Bar-code Reader, Voice Recognition, Light Pen, Touch Screen, Monitors, Printers & types – Daisy wheel, Dot Matrix, Inkjet, Laser, Line Printer, Plotter, Sound Card and Speakers.

UNIT-II: 15 mark
SOFTWARE Types of Software, Classification of System and Application Software, System Software - Operating System, Devices Drivers, Application development Packages, Word Processors, Electronic Spreadsheet, Data Base Management System, Presentation packages, Desk Top Publishing Software(DTP), Overview of languages- Machine language, Assembly language, high level languages, Types of high level languages, Generation of languages, Commands , Utility Program-Disk Cleanup Utility, Desk backup Utility, Antivirus Utilities, Customized Application Software.

UNIT-III: 15 mark
 NUMBER SYSTEM Binary, Octal, Decimal, Hexadecimal and Conversion between Number Systems, ASCII Codes, Unicode.

UNIT-IV: 15 mark
MICROPROCESSOR Basic concepts, Clock speed (MHz, GHz), 16 bit, 32 bit, 64 bit processors: Types: CISC, RISC, Concepts of System Buses, Address Bus, Data Bus, Concepts of Accumulator, Instruction Register, Program counter, Commonly used CPUs and CPU related Terminologies: Intel Pentium Series, Intel Celeron, Cyrix, AMD Series, Xeon, Intel Mobile, Mac Series, CPU Cache, Concept of heat sink and CPU fan, Motherboard; Single, Dual and Multiple Processors.

UNIT-V: 15 mark
NETWORKING & INTERNET
What is network? Need for networking, Evolution of networking, Types of networks, Data Communication & Terminologies, Network Topologies, Network Devices-Modem, Ethernet Card, Hub, Switch, Repeater, Bridge, Router, Gateway; Switching Techniques, Transmission Media, Communication Protocols, Security Concepts. Basic concept of Internet, World Wide Web, Web Browser, Web Server, Web Sites , Web Pages, URL, Domain Names, Hyper Text Mark Up Languages, Internet Address, Electronic Mail, Internet Service Provider, Search Engines.

Practical:

Part A: Hardware

1. Identification of the peripherals of a computer, components in a CPU and their functions.
2. Assembling and disassembling the system hardware components of personal computer.
3. Basic Computer Hardware Trouble shooting.
4. LAN and WiFi Basics.
5. Operating System Installation – Windows OS, UNIX/LINUX, Dual Booting.
6. Installation and Uninstallation of Software – Office Tools, Utility Software (like Anti-Virus, System Maintenance tools); Application Software - Like Photo/Image Editors, Audio Recorders/Editors, Video Editors ...); Freeware, Shareware, Payware and Trialware; Internet Browsers, Programming IDEs,
7. System Configuration – BIOS Settings, Registry Editor, MS Config, Task Manager, System Maintenance, Third-party System Maintenance Tools

Part B: Software

1. Activities using Word Processor Software
2. Activities using Spreadsheets Software.
3. Activities using Presentation Software
4. Activities involving Multimedia Editing (Images, Video, Audio ...)
5. Tasks involving Internet Browsing
6. Flow charts: Installation and using of flowgarithms software for different arithmetic tasks like sum, average, product, difference, quotient and remainder of given numbers, calculate area of Shapes (Square, Rectangle, Circle and Triangle), arrays and recursion

Text Books:

1. Basics of Information Technology-Sumita Arora,Dhanpat Rai & Co (Pvt.) Ltd.,New Delhi.
2. Foundation of Information Technology – D.S. Yadav, New Age International Publisher.
3. V. Raja Raman, "Introduction to Computers", PHI, 1998.
4. Alex Leon & Mathews Leon, "Introduction to Computers", Vikas Publishing House, 1999.
5. Norton Peter, "Introduction to computers", 4th Ed., TMH, 2001.

References:

1. Computer Applications in Management: Ritendra Goel, D.N. Kakkar, New Age International Publishers Ltd.
2. Computer Technology and Programming: H.K. Gundurao, N.S. Manjunath, M.N. Nachappa, Himalaya Publishing House.
3. Computer Applications in Management: Sanjay Saxena, Prabhapreet Chopra, Vikas Publishing House Pvt. Ltd.
4. Introduction to Computer Science: ITL Education Solutions Ltd. Pearson Education S.K.Basandra, "Computers Today ", Galgotia Publications. Alexis Leon & Mathews Leon, "Fundamentals of Information technology ", Vikas Publishing House, New Delhi.

BCA102: Programming using C (General Elective Course)
Credit: 06
Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)

Workload: 4 Lecturers (Per Week), 4 Practicals (Per Week)

Course Objective
This course is designed to impart knowledge of programming using C language thereby enabling the students to induce thinking and develop logics which will help them to create programs, applications in C. Learning the fundamentals of C programming constructs shall help the students easily switch over to any other languages in future.

Course Learning Outcomes
On successful completion of the course, students will be able to:
1. Understand the art of programming.
2. Solve programming problems.
3. Develop applications
4. Handle external files as well as exceptions.

Detailed Syllabus

UNIT-I: 15 mark
INTRODUCTION TO C PROGRAMMING: The Basic Model of Computation, Algorithms, Flow-charts, Programming Languages, Compilation, Linking and Loading, Testing and Debugging, Documentation. Programming using C: Concept of variables, program statements and function calls from the library, data types, int, char, float etc., declarations and expressions, arithmetic operation, relational and logical operations, C assignment statements, extension of assignment of the operations. C primitive input output functions, C Statements,

UNIT-II: 15 mark
CONDITIONAL STATEMENTS AND LOOPS : Decision making within a program, Conditions, Relational Operators, Logical Connectives, if statement, if-else statement, Loops: while loop, do while, for loop, Nested loops, Infinite loops, Switch statement, structured Programming .

UNIT-III: 15 mark
FUNCTIONS Top-down approach of problem solving, Modular programming and functions, Standard Library of C functions, Prototype of a function: Formal parameter list, Return Type, Function call, Block structure, Passing arguments to a Function: call by reference, call by value, Recursive Functions, arrays as function arguments. Storage Classes: Scope and extent, Storage Classes in a single source file: auto, extern and static, register, Storage Classes in a multiple source files: extern and static

UNIT-IV: 15 mark
ARRAYS One dimensional arrays: Array manipulation; Searching, Insertion, Deletion of an element from an array; Finding the largest/smallest element in an array; Two dimensional arrays, Addition/Multiplication of two matrices, Transpose of a square matrix; Null terminated strings as array of characters, Standard library string functions Structures and Unions , Structure variables, initialization, structure assignment, nested structure, structures and functions, structures and arrays: arrays of structures, structures containing arrays, unions.

UNIT-V: 15 mark

POINTERS Address operators, pointer type declaration, pointer assignment, pointer initialization, pointer arithmetic, functions and pointers, Arrays and Pointers, pointer arrays, pointers and structures, dynamic memory allocation. File Processing: Concept of Files, File opening in various modes and closing of a file, Reading from a file, writing onto a file.

Text Books:
1. E. Balagurusamy, "Programming with ANSI-C", Fourth Edition,2008, Tata McGraw Hill.
2. Venugopal K. R and Prasad S. R, "Mastering 'C'", Third Edition, 2008, Tata McGraw Hill.
3. B.W. Kernighan & D. M. Ritchie, "The C Programming Language", Second Edition, 2001, Pearson Education.

References:
1. Byron S Gottfried "Programming with C" Second edition, Tata McGrawhill, 2007 (Paper back)
2. R.G. Dromey, "How to solve it by Computer", Pearson Education, 2008.
3. Yashwant Kanetkar,, "Let us C", BPB Publications, 2007.
4. Hanly J R & Koffman E.B, "Problem Solving and Programm design in C", Pearson Education, 2009.

Practical for C Programming
1. To find the sum of square of natural number with 20
2. To print 50 odd number starting from 1.
3. To find factorial of a number.
4. To determine whether a number is prime or not.
5. To find the largest of three numbers.
6. To check whether the character entered is an alphabet or a number
7. To generate first ten elements of Fibonacci series
8. To find the roots of a quadratic equation
9. To print perimeter and area of a circle.
10. To calculate the net salary of an employee based on basic pay, DA, TA and PF.
11. To create a menu driven arithmetic calculator to perform basic arithmetic operations on the basis of user operator choice.
12. To check whether the entered character is a vowel or a consonant.
13. To accept 10 numbers from user and print largest number entered.
14. To print a Fahrenheit-to-Celsius conversion and vice vers
15. To evaluate the power series $e^x = 1 + x + x^2/2! + \ldots + x^n/n!$, $0 < x < 1$
16. To print the following

```
*
* *
* * *
* * * *
```

17. Write a program to display ASCII value of a character
18. Write a program to check whether a number is perfect or not
19. Write a program to find out the biggest of three number using nested if
20. A company insures its drivers if either of the following conditions are satisfied • Driver is married. • Driver is an unmarried, male and above 30 years of age. • Driver is unmarried, female and above 25 years of age. Write a program to decide if a driver is to be insured using logical operators.

21. Write a program to read a list of positive integers terminated by -1 and display the odd and even numbers separately and also their respective counts
22. Write a program to read values of n and x and print the value of y using switch case where a. y=n+x when n=1 b. y=1+x/n when n=2 c. y= n+3x when n=3 d. y=1+nx
23. Write a program to n values of sales and then calculate the commission on sales amount where the commission is calculated as follows: a. If sales <= Rs.500, commission is 5%. b. If sales > 500 but <= 2000, commission is Rs 35 plus 10% above Rs 500. c. If sales > 2000 but <= 5000, commission is Rs 185 plus 12% above Rs.2000. d. If sales > 5000, commission is 12.5
24. Write a program to find out minimum, maximum, sum and average of n numbers without using array
25. Program to find mean and standard deviation (SD) for a set of n numbers without using array.
26. Write a program to print the digits of a number in words. (eg. if a number 841 is entered through the keyboard your program should print "Eight Four One".
27. Write a program to print the PASCAL Triangle up to the n-th row where n is an input to the program
28. Write a function to return the HCF of two positive integers. Write a main function to read two positive integers and print their HCF and LCM by using the above function
29. Write a program to convert a decimal number into binary number using function
30. Write a program to display the result of sine series using function.
31. Write a program to find the sum of the following series $1+x-x^3/3!+x^5/5!-x^7/7!+ \ldots$ corrected up to the 3 decimal place.
32. Write a program to read n numbers in a sorted array and insert a given element in a particular position
33. Write functions to compute the factorial of a number using both recursive and non-recursive procedure.
34. Write a program to Multiply two matrices using function
35. Write a program to display the upper Triangle and lower Triangle of a given square matrix using function
36. Write a function to check if a given square matrix is symmetric or not. Write a main function to implement it.
37. Write a program to read a m x n matrix and calculate the Row sum and Column Sum of the matrix
38. Write a function to read in an integer and print the representation of the number using the sign and magnitude representation scheme using 8 bits. The program should check for overflow/under flow conditions. The left most bit is to be used as the sign bit.
39. Write a program to merge two sorted arrays.
40. Write a program to implement selection sort using function.
41. Write a program to count the number of vowels in a string.
42. Write a program to concatenate two strings using function (without using library function).
43. Write a program to convert a string from upper case to lower case and vice versa
44. Write a program to swap two numbers using function (pass the pointers).
45. Write a program to sort n number of strings in ascending order using pointer
46. Write a program using pointers to copy a string to another string variable (without using library function).

47. Declare a structure of a student with details like roll number, student name and total marks. Using this, declare an array with 50 elements. Write a program to read details of n students and print the list of students who have scored 75 marks and above
48. Write a program to read a text file and count the number of vowels in the text file
49. Write a program to copy a text file to another file.

<div align="center">

BCA203: Programming Using Python (General Elective Course)
Credit: 06
Total Marks: 100 Marks(Theory: 75 Marks, Internal Assessment: 25 Marks)
Workload: 4 Lecturers (Per Week), 4 Practicals (Per Week)

</div>

Course Objective
This course is designed to introduce the student to the basics of programming using Python. The course covers the topics essential for developing well documented modular programs using different instructions and built-in data structures available in Python.

Course Learning Outcomes
On successful completion of the course, students will be able to:
1. Develop, document, and debug modular python programs to solve computational problems.
2. Select a suitable programming construct and data structure for a situation.
3. Use built-in strings, lists, sets, tuples and dictionary in applications.
4. Define classes and use them in applications.
5. Use files for I/O operations.

Detailed Syllabus

UNIT-I: 10 mark
Introduction to Programming using Python: Structure of a Python Program, Functions, Interpreter shell, Indentation. Identifiers and keywords, Literals, Strings, Basic operators (Arithmetic operator, Relational operator, Logical or Boolean operator, Assignment Operator, Bit wise operator).

UNIT-II: 10 mark
Building blocks of Python: Standard libraries in Python, notion of class, object and method.

UNIT-III: 15 mark
Creating Python Programs: Input and Output Statements, Control statements:-branching, looping, Exit function, break, continue and pass, mutable and immutable structures. Testing and debugging a program

UNIT-IV: 15 mark
Built-in data structures: Strings, lists, Sets, Tuples and Dictionary and associated operations. Basic searching and sorting methods using iteration and recursion.

UNIT-V: 15 mark
Visualization using 2D and 3D graphics: Visualization using graphical objects like Point, Line, Histogram, Sine and Cosine Curve, 3D objects

UNIT-VI: 10 mark

Exception Handling and File Handling: Reading and writing text and structured files, Errors and Exceptions.

Text Books:
1. Downey, A.B., (2015), *Think Python–How to think like a Computer Scientist*, 3rd edition. O'Reilly Media.
2. Taneja, S. & Kumar, N., (2017), *Python Programming- A Modular Approach*. Pearson Education.
3. Reema Thareja Python Programming

References:
1. Brown, M. C. (2001). The Complete Reference: Python, McGraw Hill Education.
2. Dromey, R. G. (2006), How to Solve it by Computer, Pearson Education.
3. Guttag, J.V.(2016), Introduction to computation and programming using Python. MIT Press.
4. Liang, Y.D. (2013), Introduction to programming using Python. Pearson Education

Practical
1. Execution of expressions involving arithmetic, relational, logical, and bitwise operators in the shell window of Python IDLE.
2. Write a Python function to produce the outputs such as:

a)
```
  *
* * *
* * * * *
* * *
  *
```
(b)
```
        1
      232
    34543
   4567654
  567898765
```

3. Write a Python program to illustrate the various functions of the "Math" module.
4. Write a function that takes the lengths of three sides: **side1**, **side2** and **side3** of the triangle as the input from the user using **input** function and return the area of the triangle as the output. Also, assert that sum of the length of any two sides is greater than the third side.
5. Consider a showroom of electronic products, where there are various salesmen. Each salesman is given a commission of 5%, depending on the sales made per month. In case the sale done is less than 50000, then the salesman is not given any commission. Write a function to calculate total sales of a salesman in a month, commission and remarks for the salesman. Sales done by each salesman per week is to be provided as input. Assign remarks according to the following criteria**:**
Excellent: Sales >=80000
Good: Sales>=60000 and <80000
Average: Sales>=40000 and <60000
Work Hard: Sales < 40000
6. Write a Python function that takes a number as an input from the user and computes its factorial.

7. Write a Python function to return nth terms of Fibonacci sequence

8. Write a function that takes a number with two or more digits as an input and finds its reverse and computes the sum of its digits.

9. Write a function that takes two numbers as input parameters and returns their least common multiple and highest common factor.

10. Write a function that takes a number as an input and determine whether it is prime or not.

11. Write a function that finds the sum of the n terms of the following series:

a) $1 - x^2/2! + x^4/4! - x^6/6! + \ldots x^n/n!$

b) $1 + x^2/2! + x^4/4! + x^6/6! + \ldots x^n/n!$

12. Write a Python function that takes two strings as an input from the user and counts the number of matching characters in the given pair of strings.

13. Write a Python function that takes a string as an input from the user and displays its reverse.

14. Write a Python function that takes a string as an input from the user and determines whether it is palindrome or not.

15. Write a Python function to calculate the sum and product of two compatible matrices

16. Write a function that takes a list of numbers as input from the user and produces the corresponding cumulative list where each element in the list present at index i is the sum of elements at index $j <= i$.

17. Write a function that takes **n** as an input and creates a list of n lists such that ith list contains first five multiples of i.

18. Write a function that takes a sentence as input from the user and calculates the frequency of each letter. Use a variable of dictionary type to maintain the count.

19. Write a Python function that takes a dictionary of *word:meaning* pairs as an input from the user and creates an inverted dictionary of the form meaning:list-of-words.

20. Usage of Python debugger tool-pydb and PythonTutor.

21. Implementation of Linear and binary search techniques

22. Implementation of selection sort, insertion sort, and bubble sort techniques

23. Write a menu-driven program to create mathematical 3D objects Curve, Sphere, Cone, Arrow, Ring, and Cylinder.

24. Write a program that makes use of a function to accept a list of n integers and displays a histogram.

25. Write a program that makes use of a function to display sine, cosine, polynomial and exponential curves.

26. Write a program that makes use of a function to plot a graph of people with pulse rate p vs. height h. The values of p and h are to be entered by the user.

27. Write a function that reads a file **file1** and displays the number of words and the number of vowels in the file.

28. Write a Python function that copies the content of one file to another.

29. Write a function that reads a file **file1** and copies only alternative lines to another file **file2**. Alternative lines copied should be the odd numbered lines.

BCA512: Java Programming (General Elective Course)
Credit: 06
Total Marks: 100 Marks(Theory: 75 Marks, Internal Assessment: 25 Marks)
Workload: 4 Lectures (Per Week), 4 Practicals (Per Week)

Course Objective

This course adds to the basic programming language skills acquired by the student in earlier semesters. The students are exposed to the advanced features available in Java such as exception handling, file handling, interfaces, packages and GUI programming.

Course Learning Outcomes
On successful completion of the course the student will be
1. Implement Exception Handling and File Handling.
2. Implement multiple inheritance using Interfaces.
3. Logically organize classes and interfaces using packages.
4. Use AWT and Swing to design GUI applications.

Detailed Syllabus

UNIT-I: 10 mark
Review of Object Oriented Programming and Java Fundamentals: Structure of Java programs, Classes and Objects, Data types, Type Casting, Looping Constructs.

UNIT-II: 10 mark
Interfaces Interface basics; Defining, implementing and extending interfaces; Implementing multiple inheritance using interfaces Packages Basics of packages, Creating and accessing packages, System packages, Creating user defined packages,multithreading.

UNIT-III: 10 mark
Exception handling using the main keywords of exception handling: try, catch, throw, throws and finally; Nested try, multiple catch statements, creating user defined exceptions

UNIT-IV: 15 mark
File Handling Byte Stream, Character Stream, File I/O Basics, File Operations

UNIT-V: 15 mark
AWT and Event Handling: The AWT class hierarchy, Events, Event sources, Event classes, Event Listeners, Relationship between Event sources and Listeners, Delegation event model, Creating GUI applications using AWT.

UNIT-VI: 15 mark
Swing Introduction to Swing, Swing vs. AWT, Hierarchy for Swing components, Creating GUI
Applications using Swing.

Practical
1. Design a class Complex having a real part (x) and an imaginary part (y). Provide methods to perform the following on complex numbers:
   a. Add two complex numbers.
   b. Multiply two complex numbers.
   c. . toString() method to display complex numbers in the form: x + iy
2. Create a class TwoDim which contains private members as x and y coordinates in package P1. Define the default constructor, a parameterized constructor and override toString() method to display the co-ordinates. Now reuse this class and in package P2

create another class ThreeDim, adding a new dimension as z as its private member. Define the constructors for the subclass and override toString() method in the subclass also. Write appropriate methods to show dynamic method dispatch. The main() function should be in a package P.

3. Define an abstract class Shape in package P1. Inherit two more classes: Rectangle in package P2 and Circle in package P3. Write a program to ask the user for the type of shape and then using the concept of dynamic method dispatch, display the area of the appropriate subclass. Also write appropriate methods to read the data. The main() function should not be in any package.

4. Create an exception subclass UnderAge, which prints "Under Age" along with the age value when an object of UnderAge class is printed in the catch statement. Write a class exceptionDemo in which the method test() throws UnderAge exception if the variable age passed to it as argument is less than 18. Write main() method also to show working of the program.

5. Write a program to implement stack. Use exception handling to manage underflow and overflow conditions.

6. Write a program that copies content of one file to another. Pass the names of the files through command-line arguments.

7. Write a program to read a file and display only those lines that have the first two characters as '//' (Use try with resources).

8. Write a program to create an Applet. Create a frame as a child of applet. Implement mouseClicked( ), mouseEntered( ) and mouseExited( ) events for applet. Frame is visible when mouse enters applet window and hidden when mouse exits from the applet window.

9. Write a program to display a string in frame window with pink color as background.

10. Write a program to create an Applet that has two buttons named "Red" and "Blue". When button is pressed the background color of the applet is set to the color named by the button's label.

11. Create an applet which responds to KEY_TYPED event and updates the status window with message ("Typed character is: X"). Use adapter class for other two events.

12. Create an applet with two buttons labeled 'A' and 'B'. When button 'A' is pressed, it displays your personal information (Name, Course, Roll No, College) and when button 'B' is pressed, it displays your CGPA in previous semester.

13. Write a program that creates a Banner and then creates a thread to scrolls the message in the banner from left to right across the applet's window.

14. Rewrite the applet programs using Swing.

Text Books:
1. Balaguruswamy E. (2014). *Programming with JAVA: A Primer*. 5th edition. India: McGraw Hill Education
2. Schildt, H. (2018). *Java: The Complete Reference*. 10th edition. McGraw-Hill Education.

References:
1. Horstmann, C. S. (2017). *Core Java - Vol. I – Fundamentals* (Vol. 10). Pearson Education
2. Schildt, H., & Skrien, D. (2012). *Java Fundamentals - A Comprehensive Introduction*. India: McGraw Hill Education

BCA410: Web Technologies (Discipline Specific Core Course)
Credit 06
Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)
Workload: 4 Lectures (Per Week), 4 Practical (Per Week)

Course Objectives:
The course content enables students to:
1. Understand best technologies for solving web client/server problems
2. Analyze and design real time web applications
3. Use Java script for dynamic effects and to validate form input entry
4. Analyze to Use appropriate client-side or Server-side applications

Course Outcomes:
At the end of the course students are able to:
1. Choose, understand, and analyze any suitable real time web application.
2. To develop and deploy real time web applications in web servers and in the cloud.
3. Extend this knowledge to .Net platform

UNIT–I: 10 mark
Introduction to Web Design: Introduction of Internet, WWW, What is Website? How the Website Works?, Web pages, Front End, Back End, Client and Server Scripting Languages, Responsive Web Designing, Types of Websites (Static and Dynamic Websites) . Editors Notepad, Downloading free Editors: Notepad++, Sublime Text Editor, Making use of Editors, File creation and editing, saving

UNIT–II: 15 mark
HTML: Introduction, Basic Structure of HTML , Head Section and Elements of Head Section, Formatting Tags :Bold , Italic, Underline, Strikethrough, Div, Pre Tag Anchor links and Named Anchors ImageTag, Paragraphs, Comments, Tables : Attributes – (Border, Cellpadding, Cellspacing , height , width), TR, TH, TD, Rowspan, Colspan Lists : Ordered List , Unordered List, Definition List, Forms, Form Elements, Input types, Input Attriutes, Text Input Text Area, Dropdown Radio buttons , Check boxes ,Submit and Reset Buttons Frames : Frameset , nested Frames HTML 5 Introduction , HTML5 New Elements: Section , Nav, Article , Aside, Audio Tag, Video Tag, HTML5 Form Validations: Require Attribute, Pattern Attribute , Autofocus Attribute, email, number type, date type , Range type. Introduction to CSS, Types of CSS, CSS Selectors : Universal Selector ,ID , Tag Selector, Class Selector, Sub Selector, Attribute Selector, Group Selector, CSS Properties : Back Ground properties , Block Properties , Box properties , List properties , Border Properties , Positioning Properties, CSS Lists CSS Tables, CSS Menu Design CSS Image Gallery,

UNIT-III: 15 mark
Introduction to CSS: Types of CSS, CSS Selectors : Universal Selector ,ID , Tag Selector, Class Selector, Sub Selector, Attribute Selector, Group Selector, CSS Properties : Back Ground properties , Block Properties , Box properties , List properties , Border Properties,Positioning Properties, CSS Lists CSS Tables, CSS Menu Design CSS Image Gallery.
Web Site Development using W3.CSS Framework, W3.CSS Intro, W3.CSS Colors, W3.CSS Containers, W3.CSS Panels, W3.CSSBorders, W3.CSSFonts, W3.CSS Text, W3.CSS Tables, W3.CSS List, W3.CSSImages, W3.CSS Grid

UNIT–IV: 10 mark

Javscript and Angular Js Introduction to Client Side Scripting Language, Variables in Java Script, Operators in JS, Conditions Statements, JS Popup Boxes, JS Events, Basic Form Validations in JavaScript.

UNIT-V : 15 mark

PHP Programming: Introducing PHP: Creating PHP script, Running PHP script. Working with Variables and constants: Using variables, Using constants, Data types, Operators. **Controlling program flow**: Conditional statements, Control statements, Arrays, functions. Working with forms.

UNIT-VI: 10 mark

Web Publishing and Browsing Overview, SGML, Web hosting Basics, HTML, CGL, Documents Interchange Standards, Components of Web Publishing, Document management, Web Page Design Consideration and Principles, Search and Meta Search Engines, WWW, Browser, HTTP, Publishing Tools.

Text Books:
1. ITL Education Solution Limited, Introduction to Information Technology, Pearson Education, 2012DT Editorial Services, HTML 5 Black Book (Covers CSS3, JavaScript, XML, XHTML, AJAX, PHP, jQuery), Second Edition, Dreamtech Publisher, 2016
2. Web Technologies, Uttam Roy, OXFORD University press
3. Web programming with HTML, XHTML and CSS, 2e, Jon Duckett, Wiley Achyut Godbole,Atul Kahate, Web Technologies, Third Edition,Mc Graw Hill Education.
4. O level Web design and Publishing bpb publication.

Reference Books:
1. Web programming Bai, Michael Ekedahl, CENAGE Learning, India edition.
2. An Introduction to Web Design + Programming, Paul S.Wang, India Edition Sharma &Sharma, "Developing E-Commerce Sites", Addison Wesley Ivan Bayross, "Web Technologies Part II", BPB Publications

Practical (Web Technologies)
1. Create an HTML file (e.g. first_page.html) that specifies a page that contains a heading and two paragraphs of text. As the texts in the heading and paragraphs you can use any texts you like.
2. Write a HTML program to design a form which should allow to enter your personal data (Hint: make use of text field, password field, e-mail, lists, radio buttons, checkboxes, submit button)
3. Write HTML Code to demonstrate the use of Anchor Tag for the Following:-
   a. Creating a web link that opens in a new window.
   b. Creating a web link that opens in the same window
   c. Reference within the same html document.
   d. Reference to some image.
   e. Making an image a hyperlink to display second imag
4. Create an html page with following specifications**.** Title should be about my City. Place your City name at the top of the page in large text and in blue color. Add names

of landmarks in your city each in a different color, style and typeface. One of the landmark, your college name should be blinking. Add scrolling text with a message of your choice

5. Create an html page with 7 separate lines in different colors. State color of each line in its text.
6. Create an html page containing the polynomial expression as follows : $a^0 + a^1x + a^2x^2 + a^3x^3$
7. Write a HTML code to generate following output

First Frame: Name and address		
Second frame Bulleted list of qualifications		Third frame Links to favourite sites
Fourth frame Scrolling message	Fifth frame Blinking reminders	Sixth frame Image

8. Create a HTML Document with JavaScript code that has three Textboxes and a button. The details should be accepted using textboxes are principal, rate of interest, and duration in years. When user clicks the OK Button a message box appears showing the simple interest of principal amount.
9. Write a HTML Script to insert a hyperlink. Create a hyperlink in html which when clicked links to www.google.com in a new window. Create a HTML file which displays three images at LEFT, RIGHT and CENTER respectively in the browser.
10. Create table with ROWSPAN and COLSPAN attribute of TABLE in HTML (Prepare timetable of your class). Include CELLSPACING & CELL PADDING.
11. Create a web page, divide the web page into four frames. In one frame create three links that will display different HTML forms in the remaining three frames respectively.
12. Write a program in Java Script to print factorial
13. With CSS use the shorthand background property to set background image to eg."xyz.png", show it once, in the top right corner.
14. Write a program in javascript to generate series of prime numbers. `Write a JavaScript program to display the current day and time in the following format.
    a. Sample Output: Today is: Tuesday.
    b. Current time is: 10 PM: 30:38
15. Write a program to sum and multiply of two numbers using JavaScript.
16. Write a program to redirect, popup and print function in JavaScript.
17. Create your first "Hello world" application in AngularJS.
18. Write a code to display the words "AngularJS" in both text format and in a text box when the page is viewed in the browser.
19. Create a sample form program that collects the first name, last name, email, user id, password and confirms password from the user. All the inputs are mandatory and email address entered should be in correct format. Also, the values entered in the password and confirm password textboxes should be the same. After validating using JavaScript, in output display proper error messages in red color just next to the textbox where there is an error.

BCA825: Introduction to Multimedia
Credit: 06
Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)
Workload: 4 Lectures (Per Week), 4 Practical (Per Week)

Course Objective
This course aims to introduce the fundamental elements of multimedia. It will provide an understanding of the fundamental elements in multimedia. The emphasis will be on learning the representations, perceptions and applications of multimedia. Software skills and hands on work on digital media will also be emphasized. On completion of the subject, the students will understand the technologies behind multimedia applications and master the skills for developing multimedia projects.

Learning Outcomes
 After successfully completing the module student should be able to:
1. Summarize the key concepts in current multimedia technology.
2. Create quality multimedia software titles.

Detailed Syllabus

UNIT-I: 10   mark

Introduction to Multimedia :What is multimedia, Components of multimedia, Web and Internet multimedia applications, Transition from conventional media to digital media.

UNIT-II: 10   mark

Computer Fonts and Hypertext: Usage of text in Multimedia, Families and faces of fonts, outline fonts, bitmap fonts International character sets and hypertext, Digital fonts techniques.

UNIT-III: 10   mark

Audio fundamentals and representations
Digitization of sound, frequency and bandwidth, decibel system, data rate, audio file format, Sound synthesis, MIDI, wavetable, Compression and transmission of audio on Internet, Adding sound to your multimedia project, Audio software and hardware.

UNIT-IV: 15   mark

Image fundamentals and representations Colour Science ,Colour, Colour Models, Colour palettes, Dithering, 2D Graphics, Image Compression and File Formats :GIF, JPEG, JPEG 2000, PNG, TIFF, EXIF, PS, PDF, Use of image editing software, White balance correction, Dynamic range correction, Gamma correction, Photo Retouching.

UNIT-V: 15   mark

Video and Animation Video Basics , How Video Works, Broadcast Video Standards, Analog video, Digital video, Video Recording and Tape formats, Shooting and Editing Video (Use Adobe Premier for editing), Video Compression and File Formats. Video compression

based on motion compensation, MPEG-1, MPEG-2, MPEG-4, MPEG-7, MPEG-21, Animation: Cell Animation, Computer Animation, Morphing.

UNIT-VI: **15   mark**

Multimedia Authoring Multimedia Authoring Basics, Some Authoring Tools, Macromedia Director & Flash. The aim of the syllabus is to provide orientation as regard to uses of Multimedia. Practical Assignments may be handled using Multimedia tools, such as Flash, Dreamweaver, Photoshop etc. or any other open source multimedia tools.

RECOMMENDED BOOKS
1. Tay Vaughan, "Multimedia making it work", Tata McGraw-Hill, 2008.
2. Rajneesh Aggarwal & B. B Tiwari, "Multimedia Systems", Excel Publication, New Delhi, 2007.
3. Li & Drew, "Fundamentals of Multimedia, Pearson Education, 2009.

SUPPLEMENTARY READING
1. Parekh Ranjan, "Principles of Multimedia", Tata McGraw-Hill, 2007
2. Anirban Mukhopadhyay and Arup Chattopadhyay, "Introduction to Computer Graphics and Multimedia", Second Edition, Vikas Publishing House.

Practical

The aim of the practical is to provide orientation as regard to uses of Multimedia. Practical Assignments may be handled using Multimedia tools, such as Flash, Dreamweaver, Photoshop etc. or any other open source multimedia tools.

99