



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

COURSE STRUCTURE
&
DETAILED SYLLABUS
For
BACHELOR OF COMPUTER APPLICATIONS

MANIPUR UNIVERSITY
CANCHIPUR, IMPHAL

Course Structure (Draft)

Model A

Bachelor of Computer Application is a 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) English/MIL | BCASE-123A/B/C-(4) | VAC-101(2) | 24 |
| | BCA102(6) | | | | | VAC-102(2) | |
| II | BCA203(6) | | | AECC-2 (4) Environmental Sc. | 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) | | BCA425(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) | BCA525(6) 3(6) | | | VAC-507(2) | 26 |
| | BCA512(6) | | | | | | |
| VI | BCA613(6) | BCA620A/B/C(6) | BCA625(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) | BCA725(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 accountancy |
| 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 Modelling and

SimulationDSE-3 (Choose any one)

- BCA721A Machine 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

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

SEC-1: (Choose any one)

- a) BCASE123A Mathematical Foundation
- b) BCASE123B Office Automation Tools
- c) BCASE123C PC Hardware and maintenance

SEC-2: (Choose any one)

- a) BCASE224A Linux Environment
- b) BCASE224B Object oriented programming using C++
- c) BCASE224C Introduction to Multimedia

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 | III | Fundamentals of Information Technology |
| 2 | BCA425 | IV | Business Organization & Management |
| 3 | BCA525 | V | Organizational Behavior |
| 4 | BCA625 | VI | Operation Research |
| 5 | BCA725 | VII | Introduction to Web designing |
| 6 | BCA817 | VIII | Artificial Intelligence |

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 | 4 | Computer System Architecture | 75 | 25 | |
| | BCA204 P | Practical | 2 | -do- | 35 | 15 | |
| | BCASE-224-A/B/C | Theory and Practical | 4 | Linux Environment/ Object oriented programming using C++/ Introduction to Multimedia | 35 | 15 | |
| Exit Option with Certificate | | | | | | | |
| III | BCA305 | Theory | 4 | Data Structure using C | 75 | 25 | |
| | BCA305 P | Practical | 2 | -do- | 35 | 15 | |
| | BCA306 | Theory | 4 | Operating System | 75 | 25 | |
| | BCA306 P | Practical | 2 | -do- | 35 | 15 | |
| | BCA307 | Theory | 5 | Discrete Mathematics | 75 | 25 | |
| | BCA307 T | Tutorial | 1 | -do- | | | |
| | BCA101 | Theory | 4 | Fundamentals of information Technology | 75 | 25 | |
| | BCA101 P | 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 | |
| | BCA425 | Theory | 5 | Business Organization & Management | 75 | 25 | |
| | BCA425 | Tutorial | 1 | -do- | | | |
| Exit Option with Diploma | | | | | | | |
| | BCA511 | Theory | 5 | Computer Networks | 75 | 25 | |
| | BCA511 | Tutorial | 1 | -do- | | | |
| | BCA512 | Theory | 4 | Java Programming | 75 | 25 | |
| | BCA512 | Practical | 2 | -do- | 35 | 15 | |

| | | | | | | | |
|---|-----------------|------------------------|-----|---|----|----|--|
| V | 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 | |
| | BCA525 | Theory | 5 | Organizational Behavior | 75 | 25 | |
| | BCA525 | Tutorial | 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/Modelling and Simulation | 75 | 25 | |
| | BCA620A/B/ C | Practical/ Tutorial | 2/1 | -do- | 35 | 15 | |
| | BCA625 | Theory | 5 | Operation Research | 75 | 25 | |
| | BCA625 | Tutorial | 1 | -do- | | | |
| Exit option with Bachelor of Computer Applications | | | | | | | |
| VII | 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 | |
| | BCA721A/B/ | Theory | 4/5 | Machine learning/ Data Analysis and Visualization/ Fundamentals of Data Science | 75 | 25 | |
| | BCA721A/B/ | Practical/Tutorial | 2/1 | -do- | 35 | 15 | |
| | BCA725 | Theory | 4 | Introduction to Web designing | 75 | 25 | |
| | BCA725 | 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 | |
| | BCA817 | Theory | 4 | Artificial Intelligence | 75 | 25 | |
| | BCA817 | 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 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

15M

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: SOFTWARE

15M

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: NUMBER SYSTEM

15M

Binary, Octal, Decimal, Hexadecimal and Conversion between Number Systems, ASCII Codes, Unicode.

UNIT-IV: MICROPROCESSOR

15M

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: NETWORKING & INTERNET

15M

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.

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, Prabhpreet 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 Informationtechnology “, 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 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.
4. Develop applications
3. Handle external files as well as exceptions.

Detailed Syllabus**UNIT-I:****INTRODUCTION TO PROGRAMMING****15M**

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**15M****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**15M****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**15M****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**15M****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 versa
15. To evaluate the power series $e^x = 1 + x + \frac{x^2}{2!} + \dots + \frac{x^n}{n!}$, $0 < x < 1$
16. To print the following


```

*
* *
* * *
* * * *

```
17. To print the following


```

1
2 2
3 3 3
4 4 4 4

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

21. 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.
22. 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.
23. 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. $y=1+nx$
24. 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 \leq 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%
25. Write a program to find out minimum, maximum, sum and average of n numbers without using array.
26. Program to find mean and standard deviation (SD) for a set of n numbers without using array.
27. 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”.)
28. Write a program to print the PASCAL Triangle up to the n-th row where n is an input to the program.
29. 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.
30. Write a program to convert a decimal number into binary number using function.
31. Write a program to display the result of sine series using function.
32. Write a program to find the sum of the following series $1+x-x^3/3!+x^5/5!-x^7/7!+\dots$ corrected up to the 3 decimal place.
33. Write a program to read n numbers in a sorted array and insert a given element in a particular position
34. Write functions to compute the factorial of a number using both recursive and non-recursive procedure.
35. Write a program to Multiply two matrices using function
36. Write a program to display the upper Triangle and lower Triangle of a given square matrix using function.
37. Write a function to check if a given square matrix is symmetric or not. Write a main function to implement it.
38. Write a program to read a m x n matrix and calculate the Row sum and Column Sum of the matrix
39. 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.
40. Write a program to merge two sorted arrays.
41. . Write a program to implement selection sort using function.
42. Write a program to count the number of vowels in a string.
43. Write a program to concatenate two strings using function (without using library function).
44. Write a program to convert a string from upper case to lower case and vice versa.
45. Write a program to swap two numbers using function (pass the pointers).
46. Write a program to sort n number of strings in ascending order using pointer.
47. Write a program using pointers to copy a string to another string variable (without using library function).

48. 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.
49. Write a program to read a text file and count the number of vowels in the text file.
50. Write a program to copy a text file to another file.

BCA203: Programming Using Python (Discipline 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 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****10M**

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**10M**

Building blocks of Python: Standard libraries in Python, notion of class, object and method.

UNIT-III**15M**

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**15M**

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

UNIT-V**15M**

Visualization using 2D and 3D graphics: Visualization using graphical objects like Point, Line, Histogram, Sine and Cosine Curve, 3D objects

UNIT-VI**10M**

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.

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

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 $j \leq i$.
17. Write a function that takes n as an input and creates a list of n lists such that i th 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.

BCA204: Computer System Architecture (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 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

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

1. Design Combinational Circuits using basic building blocks. Simplify these circuits using Boolean algebra and Karnaugh maps. Differentiate between combinational circuits and sequential circuits.
2. Represent data in binary form, convert numeric data between different number systems and perform arithmetic operations in binary.
3. Determine various stages of instruction cycle and describe interrupts and their handling.
4. Explain how CPU communicates with memory and I/O devices.
5. Simulate the design of a basic computer using a software tool

Detailed Syllabus**UNIT-I 10M**

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 15M

Digital Components: Half Adder, Full Adder, Decoders, Multiplexers, Registers and Memory Units

UNIT-III 10M

Data Representation and Basic Computer Arithmetic: Number system, complements, fixed and floating point representation. Alphanumeric representation. Addition, subtraction.

UNIT-IV 15M

Basic Computer Organization and Design: Common Bus system, instruction codes, instruction format, instruction set completeness, Sequence Counter, timing and control, instruction cycle, memory reference instructions and their implementation using arithmetic, logical, program control, transfer and input output micro-operations, interrupt cycle.

UNIT-V 15M

Central Processing Unit: Micro programmed Control vs Hardwired Control, lower level programming languages, Instruction format, accumulator, general register organization, stack organization, zero-address instructions, one-address instructions, two-address instructions, threeaddress instructions, Addressing Modes, RISC, CISC architectures, pipelining and parallel processing.

UNIT-VI 10M

Memory Organization and Input-Output Organization: Input-Output Organization: Peripheral Devices, I/O interface, I/O vs. Memory Bus, Programmed I/O, Interrupt-Driven I/O, Direct Memory Access

Practical

(Use Simulator – CPU Sim 3.6.9 or any higher version for the implementation)

1. Create a machine based on the following architecture :

| Registers | | | | | | |
|-----------|------|------|------|------|-------|-------|
| IR | DR | AC | AR | PC | I | E |
| 0 15 | 0 15 | 0 15 | 0 11 | 0 11 | 1 bit | 1 Bit |

| | | | |
|---|--------------------------------|---------------------|--|
| Memory 4096 words 8 bits per word | Instruction format 0 3 4 15 | | |
| | O pc od e | A dd re ss | |

Basic Computer Instructions

| Memory Reference | Register Reference | | | |
|------------------|--------------------|---------------------|------|------|
| Symbol | Hex | Symbol | Hex | |
| AND | 0xxx | Direct Addressing | CLA | E800 |
| ADD | 2xxx | CLE | E400 | |
| LDA | 4xxx | CMA | E200 | |
| STA | 6xxx | CME | E100 | |
| BUN | 8xxx | CIR | E080 | |
| CIL | E040 | | | |
| ISZ | Cxxx | INC | E020 | |
| AND_I | 1xxx | Indirect Addressing | SPA | E010 |
| ADD_I | 3xxx | SNA | E008 | |
| LDA_I | 5xxx | SZA | E004 | |
| STA_I | 7xxx | SZE | E002 | |
| BUN_I | 9xxx | HLT | E001 | |
| ISZ_I | Dxxx | | | |

Refer to Chapter-5 of reference 1 for description of instructions.

Design the register set, memory and the instruction set. Use this machine for the assignments of this section.

2. Create a Fetch routine of the instruction cycle.
3. Write an assembly program to simulate ADD operation on two user-entered numbers.
4. Write an assembly program to simulate SUBTRACT operation on two user-entered numbers.
5. Write an assembly program to simulate the following logical operations on two userentered numbers.
 1. AND
 2. OR
 3. NOT
 4. XOR
 5. NOR
 6. NAND
6. Write an assembly program to simulate MULTIPLY operation on two user-entered numbers.
7. Write an assembly program for simulating following memory-reference instructions.
 1. ADD

- 2. LDA
- 3. STA
- 4. BUN
- 5. ISZ

8. Write an assembly language program to simulate the machine for following register reference instructions and determine the contents of AC, E, PC, AR and IR registers in decimal after the execution:

- 1. CLA
- 2. CMA
- 3. CME
- 4. HLT

9. Write an assembly language program to simulate the machine for following register reference instructions and determine the contents of AC, E, PC, AR and IR registers in decimal after the execution:

- 1. INC
- 2. SPA
- 3. SNA
- 4. SZE

10. Write an assembly language program to simulate the machine for following register reference instructions and determine the contents of AC, E, PC, AR and IR registers in decimal after the execution:

- 1. CIR
- 2. CIL

11. Write an assembly program that reads in integers and adds them together; until a negative non-zero number is read in. Then it outputs the sum (not including the last number).

12. Write an assembly program that reads in integers and adds them together; until zero is read in. Then it outputs the sum.

13. Create a machine for the following instruction format:

Instruction format

| | | | | |
|---------|----|---------|----|---|
| 15 14 | 13 | 12 | 11 | 0 |
| OP code | I | Address | | |

The instruction format contains a 3-bit opcode, a 1-bit addressing mode and a 12-bit address.

Write an assembly program to simulate the machine for addition of two numbers with I= 0 (Direct Address) and address part = 082. The instruction to be stored at address 022 in RAM, initialize the memory word with any decimal value at address 082. Determine the contents of AC, DR, PC, AR and IR in decimal after the execution.

14. Simulate the machine for the memory-reference instruction referred in above question with I= 1 (Indirect Address) and address part = 082. The instruction to be stored at address 026 in RAM. Initialize the memory word at address 082 with the value 298.

Initialize the memory word at address 298 with operand 632 and AC with 937. Determine the contents of AC, DR, PC, AR and IR in decimal after the execution.

15. The instruction format contains 3 bits of opcode, 12 bits for address and 1 bit for

addressing mode. There are only two addressing modes, $I = 0$ is direct addressing and $I = 1$ is indirect addressing. Write an assembly program to check the I bit to determine the addressing mode and then jump accordingly.

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.

Course Teaching Learning Process

- Use of ICT tools in conjunction with traditional class room teaching methods
- Interactive sessions
- Class discussions

BCA305: Data Structure using C (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 aims at developing the ability to use basic data structures like array, stacks, queues, lists, trees and hash tables to solve problems. C++ is chosen as the language to understand implementation of these data structures.

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. (RBT L2/3)
2. Write a program, choosing a data structure, best suited for the application at hand. (RBT L3/4)
3. Re-write a given program that uses one data structure, using a more appropriate/efficient data structure (RBT L4)
4. Write programs using recursion for simple problems. Explain the advantages and disadvantages of recursion.(RBT L2/L3)
5. Identify Ethical Dilemmas.

Detailed Syllabus**UNIT-I****10M**

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

UNIT-II**15M**

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**10M**

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

UNIT-IV**15M**

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**15M**

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**10M**

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++*. 2ndEdition, 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: 4 Lecturers (Per Week), 4 Practicals (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 multiprocessing, 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****15M**

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**15M**

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

UNIT-III**15M**

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**15M**

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**15M**

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 behaviour of Linux kernel including kernel version, CPU type and model. (CPU information)
3. Write a program to report behaviour 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:

1. Dhamdhare, D. M. (2006). *Operating Systems: A Concept-based Approach*. 2nd edition. Tata McGraw-Hill Education.
2. Kernighan, B. W., & Rob Pike, R. (1984). *The Unix programming environment* (Vol. 270). Englewood Cliffs, NJ: Prentice-Hall
3. Stallings, W. (2018). *Operating Systems: Internals and Design Principles*. 9th edition. Pearson Education.
4. 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 Lecturers (Per Week), 2 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****15M**

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**15M**

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

UNIT-III**15M**

Recurrence: recurrence relations, generating functions, linear recurrence relations with constant coefficients and their solution, recursion trees, Master Theorem

UNIT-IV**15M**

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**15M**

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:

- i) Distributions to study the joint behavior of two random variables.
- ii) To establish a formulation helping to predict one variable in terms of the other, i.e., correlation and linear regression.
- iii) 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: Probability Functions and Moment Generating Function 15M**

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: Univariate Discrete and Continuous Distributions 20M

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: Bivariate Distribution 20M

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: Correlation, Regression and Central Limit Theorem 20M

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 with 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 Lecturers (Per Week), 4 Practicals (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 functions
2. Model an application's data requirements using conceptual modelling tools like ER diagrams and 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.
5. Describe the notions of indexes, views, constraints and transactions.

Detailed Syllabus**UNIT-I****10M**

Introduction to databases: Characteristics of database approach, data models, database system architecture, data independence and data abstraction.

UNIT-II**10M**

Data modeling: Entity relationship (ER) modeling: Entity types, relationships, constraints, ER diagrams, EER model

UNIT-III**10M**

Relation data model: Relational model concepts, relational constraints, relational algebra.

UNIT-IV**15M**

SQL queries: SQL data definition, data types, specifying constraints, Queries for retrieval, insertion, deletion, updation, introduction to views.

UNIT-V:**15M**

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

UNIT-VI:**15M**

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

Text book:

1. Ramez 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 McGraw-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, Addison Wesley (Fifth Edition), 1994
- 4, J.D, Ullman, Principles of Database Systems. Galgotia Publishing (Second Edition), 1994
- 5, D.M. Kroenke, Database Processing : Fundamentals, Design Implementation, Prentice Hall (Fifth Edition) 1994

Practical (Database Management System)

1. 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.
7. List of names and cities of all salesmen in London with commission above 10%
8. List all customers excluding those with rating ≤ 100 unless they are located in London.
9. List all orders for more than Rs.1000 except the orders of snum <1006 of 10/03/21.
10. List all order taken on October 3rd or 4th or 6th, 2008.
11. List all customers whose names begins with a letter ' A'.
12. List all customers whose names begins with letter 'A' to 'G'.
13. List all orders with zero or NULL amount.

Practical List – 2

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

1. List all salesmen with their % of commission.
2. 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.
3. 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 is⊗rating)
5. List all in descending order of rating.
6. 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 Lecturers (Per Week), 4 Practicals (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. Integrate java and server side scripting languages to develop web applications.
3. To develop and deploy real time web applications in web servers and in the cloud.
4. Extend this knowledge to .Net platforms.

UNIT-I**15M**

HTML Common tags- List, Tables, images, forms, Frames, Links and Navigation, Image Maps CSS: Introduction, CSS Properties, Controlling Fonts, Text Formatting, Pseudo classes, Selectors, CSS for Links, Lists, Tables.

Java Script: Learning Java script: Variables, operators, Functions, Control structures, Events, Objects, Validations

UNIT-II**15M**

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-III**15M**

AJAX: Introduction, AJAX with XML Servlets: introduction to servlets, Life cycle of servlets, JSDK, The servlet API, the javax.servlet package, Reading servlet parameters and initialization parameters, The javax.servlet HTTP package, Handling Http request and responses, Using cookie, session tracking,

Introduction to JSP: The problem with servlet, the anatomy of JSP page, JSP processing, JSP application design with MVC, Tomcat server and testing tomcat, Generic dynamic content, using scripting elements implicit JSP objects.

UNIT-IV**15M**

Java Script Objects, Methods, Events and Functions, Tags, Operators, Data Types, Literals and Type Casting in JavaScript, Programming Construct, Array and Dialog Boxes, Relating JavaScript to DHTML, Dynamically Changing Text, Style, Content.

UNIT-V**15M**

JSP application development: Conditional processing display values using an expression to set an attribute, Declaring variables and methods, sharing data between JSP pages, Requests and users passing control and data between pages, Sharing sessions and application data, memory usage considerations

JDBC connectivity in JSP: Data base programming using JDBC, Studying javax.sql. package, Accessing a database from a JSP page, Application specific database actions, Deploying JAVA beans in JSP page.

Text Books :

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

Reference Books:

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

BCA511: Computer Networks (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 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****10M**

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**10M**

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**15M**

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), Datalink and MAC addressing, Ethernet, data link layer switching, point-to-point protocol.

UNIT-IV**15M**

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**15M**

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**10M**

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: 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), 2 Tutorials (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****15M**

Languages and Regular Expressions: Alphabets, string, language, basic operations on language, concatenation, union, Kleene star, Regular expressions.

UNIT-II**15M**

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**15M**

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

UNIT-IV**15M**

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**15M**

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)

Referecnes:

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.

BCA613: Programming in JAVA (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 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 10M**

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

UNIT-II 10M

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 10M

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 15M

File Handling Byte Stream, Character Stream, File I/O Basics, File Operations

UNIT-V 15M

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 15M

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:
 1. Add two complex numbers.
 2. Multiply two complex numbers.
 3. 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 a 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.

Referecnes:

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

BCA614: Financial Accounting (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)****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****15M**

Meaning and Nature of Financial Accounting, Scope of Financial Accounting, Financial Accounting & Management Accounting, Accounting concepts & convention, Accounting standards in India.

UNIT-II**20M**

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**20M**

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**20M**

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:

1. Gupta R. L., Gupta V. K., "Principles & Practice of Accountancy", Sultan Chand & Sons, 1999.
2. Khan & Jain, "Financial Accounting"
3. Maheshwari S. N., "Principals of Management Accounting", 11th Edition, Sultan Chand & Sons, 2001.
4. 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 Practicals (Per Week)****Course Objective**

This course is designed to introduce the students to design and analysis of algorithms in terms of efficiency and correctness. The course focuses on highlighting the 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****15M**

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**15M**

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**15M**

Advanced Analysis Technique: Amortized analysis

UNIT-IV**15M**

Graphs: Graph Algorithms - Breadth First Search, Depth First Search and its Applications.

UNIT-V**15M**

Greedy method: General method, application-job sequencing with dead lines, Knapsack problem, Minimum cost spanning trees, single source shortest path problem.

Practical

1. a) Implement Insertion Sort (The program should report the number of comparisons)
- b) Implement Merge Sort (The program should report the number of comparisons)
2. Implement Heap Sort (The program should report the number of comparisons)
3. Implement Randomized Quick sort (The program should report the number of comparisons)
4. Implement Radix Sort
5. Create a Red-Black Tree and perform following operations on it: i. Insert a node ii. Delete a node iii. Search for a number & also report the color of the node containing this number.

6. Write a program to determine the LCS of two given sequences
7. Implement Breadth-First Search in a graph
8. Implement Depth-First Search in a graph
9. Write a program to determine the minimum spanning tree of a graph

For the algorithms at S.No 1 to 3 test run the algorithm on 100 different inputs of sizes varying from 30 to 1000. Count the number of comparisons and draw the graph. Compare it with a graph of $n \log n$.

References

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

Additional Resources

1. Cormen, T.H., Leiserson, C.E. Rivest, R.L., & Stein, C. (2015). *Introduction to Algorithms*. 3rd edition. PHI.
2. Sarabasse & Gleder A.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**UNIT-I****15M**

Introduction: 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**15M**

Drawing and clipping primitives: Raster scan line, circle and ellipse drawing algorithms, Polygon filling, line clipping and polygon clipping algorithms

UNIT-III**15M**

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

UNIT-IV**15M**

Geometric Modeling: Polygon Mesh Representation, Cubic Polynomial curves (Hermite and Bezier).

UNIT-V**15M**

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 Practicals (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 10M**

Introduction: Introduction to artificial intelligence, background and applications, Turing test, rational agents, intelligent agents, structure, behaviour and environment of intelligent agents.

UNIT-II 15M

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

UNIT-III 15M

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 15M

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 10M

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 10M

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

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**15M**

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

UNIT-III**15M**

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**15M**

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**15M**

Software Maintenance: Management of Maintenance, Maintenance Process, Reverse Engineering, Software Reengineering, Configuration Management, Documentation.

TEXT BOOKS:

- 1.K.K.Aggarwal&YogeshSingh,“SoftwareEngineering”,2ndEd.,NewAgeInternational, 2005.
- 2.R.S.Pressman,“SoftwareEngineering–Apractitioner“sapproach”,5thEd.,McGrawHill Int. Ed., 2001.

REFERENCE BOOKS:

1. Stephen R. Schach, “Classical & Object-Oriented Software Engineering”, IRWIN, 1996.
2. James Peter, W. Pedrycz, “Software Engineering: An Engineering Approach”, John Wiley & Sons.
3. I. Sommerville, “Software Engineering”, Addison Wesley, 2002

Discipline Specific Elective Course- (DSE-1)**a) BCA519A: .NET Programming****Core Credit 06****Total Marks: 100 Marks(Theory: 75 Marks, Internal Assessment: 25 Marks)****Workload: 4 Lecturers (Per Week), 4 Practicals (Per Week)****Detailed Syllabus****UNIT-I****15M**

.Net architecture, Namespheres, Assemblies, object oriented features, memory management, interoperation with 10M, transaction in .NET, structured exception handling, code access security.

UNIT-II**15M**

VB.NET Similarities & differences with Visual Basic, windows focus, ADO.NET, working with databases, object oriented features.

UNIT-III**15M**

ASP.NET Similarities & difference with ASP, Architecture, web-form, development, XML, databases interface.

UNIT-IV**15M**

C++.NET Similarities & differences with C/C++, Creating components, window four, menus, validation, database interface.

UNIT-V**15M**

Overview of Microsoft Database Access Technology, ADO.Net, Creating a Database, ADO.Net Architecture, ADO.Net Class Libraries, Databound Control.

TEXT:

1. A. Chakraborti et. al., "Microsoft .NET framework", PHI, 2002
2. M. Reynolds et. al., ".NET Enterprise", Wrox/SPD, 2002

REFERENCES:

1. Richard Blair & Mathew Reynolds, "Beginning VB.net 2003", 3rd Edition, Wiley Dream Tech, 2003
2. Chris Willman, John Kauffman, "Beginning ASP net 1.1 with VB. NET 2003", Wiley Dream Teach, 2003
3. Chris Ullman, John Kauffman, "Beginning ASP. NET with Visual # net 2003", Wiley Dream Teach, 2003.

b) BCA519: Operational Research for Computer Science (Discipline Specific Elective Course)**Credit: 06****Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)****Workload: 4 Lecturers (Per Week), 4 Practicals (Per Week)****Detailed Syllabus****UNIT-I****Introductory Linear Algebra****10M**

System of linear equations, Matrices, Rank and Determinant of a matrix, Linearly dependent and independent vectors, Basis of a matrix.

UNIT-II**Linear programming-I****15M**

Optimization Problems, Introduction to LP Formulation, Convex sets, Extreme points, Geometry of Linear Programs, Basic feasible solutions (BFS), Neighborhoods, Local and global optima, Profitable Column, Pivoting, Simplex Algorithm with initial BFS, Graphical method.

UNIT-III**Linear programming-II****10M**

Degeneracy and Bland's Anticycling rule (Definition), Simplex Algorithm without initial BFS, Artificial variable techniques—two phase method, M-Charnes method, special cases in LPP.

UNIT-IV**Duality****10M**

Definition of the dual problem, primal-dual relationships, economic interpretation of duality, complementary slackness conditions.

UNIT-V**Transportation Models****10M**

Transportation Algorithm, Assignment model, Hungarian Method

UNIT-VI**Introduction to Queuing Models****10M**

Elements of Queuing Model, Exponential distribution, Poisson Distributions, Poisson Queuing Models, Single Server model, Multiple Server model

UNIT-VII**Introduction to Markov Chains****10M**

Introduction to Markov chains, transition probabilities, classification of states, Steady state probabilities, Absorbing states

Reference Books

- 1. G. Hadley: Linear Programming. Narosa, 2002 (reprint).**
- 2. A. Ravindran, D. T. Phillips and James J. Solberg: Operations Research-Principles and Practice, John Wiley & Sons, 2005.**
- 3. Hamdy A. Taha: Operations Research-An Introduction, Prentice Hall, 8th Edition, 2008.**
- 4. F.S. Hillier. G.J. Lieberman: Introduction to Operations Research-Concepts and Cases, 9th Edition, Tata McGraw Hill. 2010.**

c) BCA519: Computer Oriented Numerical Methods**Credit: 06****Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)****Workload: 4 Lecturers (Per Week), 4 Practicals (Per Week)****Detailed Syllabus****UNIT-I****15M**

Errors & Their Accuracy, Solutions of Algebraic and Transcendental Equations, Bisection Method, The method of false 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**20M**

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, Gauss's Central difference formulae, Interpolation with unevenly spaced points Lagrange's interpolation, Error in Lagrange's interpolation, Divided differences and their properties, Newton's general interpolation formula-interpolation by iteration, inverse interpolation.

UNIT-III**20M**

Differentiation and Integration, Numerical Differentiation, Methods based on interpolation, Nonuniform nodal points-Linear interpolation, 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, 3/8 Simpson's rule, open type integration rules. Methods based on undetermined coefficients-Newton's methods, Trapezoidal rule, Simpson's rule.

UNIT-IV**15M**

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-By S. 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

6. To find out the root of the Algebraic and Transcendental equations using Bisection method.
7. To find out the root of the Algebraic and Transcendental equations using Newton-Raphson method.
8. To implement Numerical Integration using Trapezoidal rule.
9. To implement Numerical Integration using Simpson 1/3 rule.
10. To implement Numerical Integration Simpson 3/8 rule.

Discipline Specific Elective Course- (DSE-II)

a) 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 Lecturers (Per Week), 4 Practicals (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

15M

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

15M

Working of Controlled Systems, Real-time systems with feedback loop e.g. thermostat in refrigerator, AC, etc. Connectivity models – TCP/IP versus OSI model, different type of modes using wired and wireless methodology, The process flow of an IoT application.

UNIT-III

15M

Sensors, Actuators and Microcontrollers:

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

15M

Building IoT applications: Embedded '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

15M

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

b) BCA620B: Cloud Computing (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:**

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****25M**

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, Multi-tenancy (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**15M**

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**15M**

Introduction, Storage as a service, Amazon storage services, Compute as a service Amazon elastic compute cloud (EC2), Cloud System matrix, Platform as a Service, Windows Azure, Google Apps Engine, Amazon Web services, Software as a Service CRM as a service, sales force.com 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, SAN fabric, NAS gateways.

UNIT-IV

20M

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 - Anthony T. Velte, Toby J. Velte Robert Elsenpeter, TATA McGraw-Hill, New Delhi - 2010
3. Cloud Computing: Web-Based Applications That Change the Way You Work and Collaborate Online - Michael Miller - Que 2008

References:

1. Moving to Cloud by Dinkar Sitaram, Geetha Manjunath, Publication: Syngress Elsevier Inc, 2014 (2nd Edition)
2. Cloud Computing Second Edition by Dr Kumar Saurabh, Publication Willy INDAI (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. Cloud Computing (Principles and Paradigms), Edited by Rajkumar Buyya, James Broberg, Andrzej Goscinski, John Wiley & Sons, Inc. 201

c) BCA620C: Modelling and Simulation (BC620C)**Credit: 06****Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)****Workload: 4 Lecturers (Per Week), 4 Practicals (Per Week)****Course Learning Objectives:**

1. Define the basics of simulation modeling and replicating the practical situations in organizations
2. Generate random numbers and random variates using different techniques.
3. Develop simulation model using heuristic methods.
4. Analysis of Simulation models using input analyzer, and output analyzer
5. Explain Verification and Validation of simulation model.

Learning outcome: At the end of the course, the student should know:

- 1 Models, model properties and some modeling tools:
2. Know the most important model classes.
3. Understand how to manipulate models / model approximations to change their class

Unit-I:**System models and System studies****15M**

Concept of a System, Deterministic and Stochastic Activities, Continuous and Discrete Systems, System Modeling, Types of Models, Principles used in Modeling, Corporate Model, System Design.

What is Simulation?

Technique of Simulation, The Monte-Carlo Method, Comparison of Simulation and Analytical Methods, Experimental nature of Simulation, Types of System Simulation, Numerical Computation Techniques for Continuous Models, Numerical Computation Techniques for Discrete Models, Distributed Lag Models, Cobweb Models.

Unit-II:**Continuous System Simulation****10M**

Continuous System Models, Differential Equations & Applications, Feedback Systems, Simulation of an Autopilot, Interactive systems, Real Time Systems.

Unit-III:**Concepts in Discrete Event Simulation****10M**

The Event Scheduling/Time Advance Algorithm, World Views, Manual Simulation using Event Scheduling, List Processing: Lists-Basic properties and operations, Use of arrays for List Processing, Using Dynamic Allocation and Linked Lists.

Queuing Models

Characteristics of Queuing Systems, Arrival and Service Patterns, Queue Discipline, Long Run Measures of Performance of Queuing Systems, Time-Average Number in System, Server Utilization, Costs in Queuing Problems, Steady State behavior of Infinite Population Markovian Models, Multiserver Queue: $M/M/C/\infty/\infty$.

Unit-IV:**Simulation software [7 Lectures]****10M**

Comparison of Simulation Packages with Programming Languages, Classification of Simulation Software: General Purpose vs Application Oriented Simulation Packages, Desirable Software Features: General Capabilities, H/w and S/w Requirements, Animation and Dynamic Graphics, Statistical Capabilities, General Purpose Simulation Packages, Object Oriented Simulation, Examples of Application-oriented Simulation packages, Simulation in GPSS.

Unit-V:**10M****Random Number, Non-Uniform random variate Generation and Monte-Carlo Method**

Linear Congruential Generators, Testing Random Number Generators: Empirical and Theoretical tests, Non-Uniform Random Variate Generator: Inverse Transform, Composition. Generating Continuous Random Variates: Uniform, Exponential, Gamma, and Normal. Generating Discrete Random Variates: Bernoulli, Binomial, Poisson. Monte-Carlo Method: Evaluation of Integral-Hit or Miss Method.

Unit-VI:**Analysis of Simulation Data****10M**

Identifying the Distribution with Data, Types of Simulations with respect to Output Analysis, Stochastic nature of Output Data, Measures of Performance and their Estimation: Point Estimation, Confidence-Interval Estimation, Output Analysis for Terminating Simulations: Statistical Background, Confidence-Intervals with Specified Precision, Output Analysis for Steady State Simulations, Variance-Reduction Technique-Antithetic variates.

Unit-VII:**Verification and Validation of Simulation Models****10M**

Model Building-Verification and Validation, Verification of Simulation Models, Calibration and Validation of Models: Validation of Model Assumptions, Validating Input-output Transformations, Input-Output Validation

References:

1. Raj Jain, Art of Computer Systems Performance Analysis, John Wiley and Sons, Inc, 1991
2. Sheldon M. Ross, Simulation, 4th Ed., Elsevier 2008
3. Averill M. Law and W. David Kelton, Simulation Modeling and Analysis, 3rd Ed., Tata McGraw-Hill, 2003
4. Geoffrey Gordon, System Simulation, 2nd Ed., PHI, 1987
5. Jerry Banks and John S. Carson, Barry L. Nelson, Discrete-Event System Simulation, 5th Ed., Prentice Hall, 2010
6. Narsingh Deo, System Simulation with Digital Computers, Prentice Hall of India, 1979

Discipline Specific Elective Course- (DSE-III)

a) BCA721A: Machine Learning (Discipline Specific Elective)

Credit: 06

Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)

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

Course Objective

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 the theoretical basis of SVM
4. Implement various machine learning algorithms learnt in the course.

Detailed Syllabus

UNIT-I

10M

Introduction: Basic definitions, Hypothesis space and inductive bias, Bayes optimal classifier and Bayes error, Occam's razor, Curse of dimensionality, dimensionality reduction, feature scaling, feature selection methods.

UNIT-II

15M

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

25M

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

25M

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 softwares like MABLAB/Octave or Python. For later exercises, students can create/use their own datasets or utilize datasets 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 simple 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 range, 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 other matrix operations like converting matrix data to absolute values, taking the negative of matrix values, adding/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 types 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 types of loops based on simple example/s.
11. Perform vectorized 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 dataset 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 an email 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 backpropagation - 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.

b) BCA721B: Data Analysis and Visualization (Discipline Specific Elective)**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 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****15M**

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, statsmodels

UNIT-II**15M**

Getting Started with Pandas: Arrays and vectorized 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**15M**

Data Wrangling: Hierarchical Indexing, Combining and Merging Data Sets Reshaping and Pivoting.

Data Visualization matplotlib: Basics of matplotlib, plotting with pandas and seaborn, other python visualization tools

UNIT-IV**15M**

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**15M**

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. Practicals based on NumPy ndarray
2. Practicals based on Pandas Data Structures
3. Practicals based on Data Loading, Storage and File Formats
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'Reilly Media.
2. O'Neil, C., & Schutt, R. (2013). *Doing Data Science: Straight Talk from the Frontline*. O'Reilly Media

c) BCA721C: Fundamentals of Data Sciences**Credit: 06****Total Marks: 100 Marks(Theory: 75 Marks, Internal Assessment: 25 Marks)****Workload: 4 Lecturers (Per Week), 4 Practicals (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****15M**

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, markdown, git, GitHub, R, and RStudio. 12L

UNIT-II**15M**

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 12L

UNIT-III**15M**

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". 12L

UNIT-IV**15M**

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. 12L

UNIT-V**15M**

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 a literate statistical program, compile R markdown documents using knitr and related tools, and organize a data analysis so that it is reproducible and accessible to others. 12L

Reference Books

1. Rachel Schutt, Cathy O'Neil, "Doing Data Science: Straight Talk from the Frontline" by Schrott/O'Reilly, 2013.
2. Foster Provost, Tom Fawcett, "Data Science for Business" What You Need to Know About Data Mining and Data-Analytic Thinking" by O'Reilly, 2013.
3. John W. Foreman, "Data Smart: Using data Science to Transform Information into Insight" by John Wiley & Sons, 2013.
4. Ian Ayres, "Super Crunchers: Why Thinking-by-Numbers Is the New Way to Be Smart" 1st Edition by Bantam, 2007.
5. Eric Siegel, "Predictive Analytics: The Power to Predict who Will Click, Buy, Lie, or Die", 1st Edition, by Wiley, 2013.

6. Matthew A. Russel, "Mining the Social Web: Data mining Facebook, Twitter, LinkedIn, Google+, GitHub, and More", Second Edition, by O'Reilly Media, 2013.

Fundamental of Data Sciences Lab

1. Write a program that prints 'Hello World' to the screen.
2. Write a program that asks the user for a number n and prints the sum of the numbers 1 to n
3. Write a program that prints a multiplication table for numbers up to 12.
4. Write a function that returns the largest element in a list.
5. Write a function that computes the running total of a list.
6. Write a 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

- i. Find the median age of all students under 22 years ii.

Find the median age of all students

| | |
|------|--|
| | Find the mean age of all students |
| iii. | Find the modal age for all students |
| iv. | Two more students enter the class. The age of both students is 23. What is |
| v. | 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 X , 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 ggplot2 plotting systems in R, apply basic principles of data graphics to create rich analytic graphics from available datasets.

14. Use Git / Github software to create Github account. Also, create a repository on Github

Discipline Specific Elective Course- (DSE-1V)

a) 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 1

15M

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 2

15M

Spatial Domain Filtering: Some Basic Intensity Transformation Functions, Histogram Equalization, Spatial Correlation and Convolution, Smoothing Spatial Filters: Lowpass filters, Order Statistics filters; Sharpening Spatial Filters: Laplacian filter

Unit 3

10M

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 4

15M

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

10M

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

Unit 6

10M

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

- a. Become familiar with SCILAB/MATLAB Basic commands
 - b. Read and display image in SCILAB/MATLAB
 - c. Resize given image
 - d. Convert given color image into gray-scale image
 - e. Convert given color/gray-scale image into black & white image
 - f. Draw image profile
 - g. Separate color image in three R, G & B planes
 - h. Create color image using R, G and B three separate planes
 - i. Flow control and LOOP in SCILAB
 - j. Write given 2-D data in image file
2. To write and execute image processing programs using point processing method
 - a. Obtain Negative image
 - b. Obtain Flip image
 - c. Thresholding
 - d. Contrast stretching
 3. 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
 4. To write and execute programs for image logical operations
 - a. AND operation between two images
 - b. OR operation between two images
 - c. Calculate intersection of two images
 - d. WaterMarking using EX-OR operation
 - e. NOT operation (Negative image)
 5. To write a program for histogram calculation and equalization using
 - a. Standard MATLAB function
 - b. Program without using standard MATLAB functions
 - c. C Program
 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. Use 3x3 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. John Wiley and Sons.

b) BCA822B: NetworkSecurity (DisciplineSpecific Elective)

Credit: 06

Total Marks: 100 Marks(Theory: 75 Marks, InteernalAssessment: 25 Marks)

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

CourseObjective

The courseoffers abroadoverview ofthefundamentals of information securitycoveringtopics such as error correction/detection, cryptography, steganography, malwares,This course also touches on theimplications of securityinInternetof Things (IoT).

CourseLearning Outcomes

On successfulcompletion ofthis course, astudentwillbe able to,

1. Identifythe major types of threats to information security
2. Describetheroleofcryptographyin security
3. Select appropriate error-detection and error-correction methods for anapplication
4. Discuss thestrengths and weaknesses of privateand publickeycrypto systems
5. Describemalwares and memoryexploits
6. Discuss theneed for securityinIoT

Detailed Syllabus

Unit-I

15M

Informationsecurit

Attributes of Information Security: Confidentiality, Integrity, Availability. Threats& Vulnerabilities: Unauthorized Access, Impersonation, DenialofService, MaliciousSoftware; TrapDoors, LogicBomb, TrojanHorses; Viruses, Worms&Bacteria; CryptographyBasics: Plain Text, Cipher Text, Encryption Algorithm, DecryptionAlgorithm; Requirements for Cryptography, SymmetricvsAsymmetric, BlockandStreamciphers, DES. T[1], T[2]

Unit-II

10M

PublicKeyInfrastructure&. MessageAuthentication

Public Key Cryptography Principles & Applications, Algorithms: RSA, Message Authentication: One way Hash Functions: Message Digest, MD5, SHA1. PublicKey Infrastructure: DigitalSignatures, Digital Certificates, CertificateAuthorities. T[1], T[2]

Unit-III

Network Security

15M

NetworkAttacks: BufferOverflow, IPspoofing, TCPSessionHijacking, SequenceGuessing,

NetworkScanning:

ICMP, TCPsweeps, BasicPortScans; DenialofServiceAttacks: SYNflood, Teardropattacks, land, SmurfAttacks.

IP security Architecture: Overview, Authentication header, Encapsulating Security Pay Load, combining Security Associations, Key Management. Virtual Private Network Technology: Tunnelingusing IPSEC. T[1], T[2]

Unit-IV Web Security

10M

Requirements, Secure Socket Layer, and Secure Electronic Transactions, Network Management Security: Overview of SNMP Architecture-SNMPV1, SNMPV3. Firewall Characteristics & Design Principles, Types of Firewalls: Packet Filtering Router, Application Level Gateway or Proxy, Content Filters, Bastion Host. T[1], T[2]

Practical

1. Implement the error correcting code.
2. Implement the error detecting code.
3. Implement Caesar cipher substitution operation.
4. Implement monoalphabetic and polyalphabetic cipher substitution operation.
5. Implement Playfair cipher substitution operation.
6. Implement Hill cipher substitution operation.
7. Implement rail fence cipher transposition operation.
8. Implement row transposition cipher transposition operation.
9. Implement product cipher transposition operation.
10. Illustrate the ciphertext only and known plaintext attacks.
11. Implement 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.

c) **BCA822C: Data Mining**

Credit: 06

Total Marks: 100 Marks(Theory: 75 Marks, Internal Assessment: 25 Marks)

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

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

25M

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

15M

Classification: Basic Concepts, Decision Tree Classifier: Decision tree algorithm, attribute selection measures, Nearest Neighbour Classifier, Bayes Theorem and Naive Bayes Classifier, **Model**

Evaluation: Holdout Method, Random Sub Sampling, Cross-Validation, evaluation metrics, confusion matrix.

Unit-III

15M

Association rule mining: Transaction data-set, Frequent Itemset, Support measure, Apriori Principle, Apriori Algorithm, Computational Complexity, Rule Generation, Confidence of association rule.

Unit-IV

25M

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

Q1. Create a file "people.txt" with the following data:

| Age | agegroup | height | status | yearsmarried |
|-----|----------|--------|---------|--------------|
| 21 | adult | 6.0 | single | -1 |
| 2 | child | 3 | married | 0 |
| 18 | adult | 5.7 | married | 20 |
| 22 | elderly | 5 | widowed | 2 |
| 34 | child | -7 | married | 3 |

i) Read the data from the file "people.txt".

ii) Create a rule set E that contains rules to check for the following conditions:

1. The age should be in the range 0-150.
2. The age should be greater than years married.
3. The status should be married or single or widowed.

4. If ages less than 18 the age group should be child, if ages between 18 and 65 the age group should be adult, if ages more than 65 the age group should be elderly.

iii) Check whether the ruleset E is violated by the data in the file `people.txt`. iv)

Summarize the results obtained in part (iii)

v) Visualize the results obtained in part (iii)

Q2. Perform the following preprocessing tasks on the `dirty_iris` dataset.

1. Calculate the number and percentage of observations that are complete.

2. Replace all the special values in data with NA.

3. Define these rules in a separate text file and read them.

(Use `editfile` function in R (package `editrules`). Use similar function in Python). Print the resulting constraint object.

– Species should be one of the following values: `setosa`, `versicolor` or `virginica`.

– All measured numerical properties of an iris should be positive.

– The petal length of an iris is at least 2 times its petal width.

– The sepal length of an iris cannot exceed 30 cm.

– The sepal of an iris are longer than its petals.

4. Determine how often each rule is broken (violated). Also summarize and plot the result.

Find outliers in sepal length using `boxplot` and `boxplot.stats`

Q3. Load the data from `wine` dataset. Check whether all attributes are standardized or not (mean is 0 and standard deviation is 1). If not, standardize the attributes. Do the same with `Iris` dataset. Section 2: Data Mining Techniques

Run following algorithms on 2 real datasets and use appropriate evaluation measures to compute correctness of obtained patterns:

Q4. Run Apriori algorithm to find frequent itemsets and association rules

4.1 Use minimum support as 50% and minimum confidence as 75%

4.2 Use minimum support as 60% and minimum confidence as 60%

Q5. Use Naive 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:

5.1 a) Training set = 75% Test set = 25%

b) Training set = 66.6% (2/3rd of total), Test set = 33.3%

5.2 Training set is chosen by i) hold out method ii) Random subsampling iii) Cross Validation. Compare the accuracy of the classifiers obtained.

5.3 Data is scaled to standard format.

Q6. Use Simple Kmeans, DBScan, Hierarchical clustering algorithms for clustering. Compare the performance of clusters by changing the parameters involved in the algorithms.

Recommended Datasets for Data Mining practicals

1. UCIMachine 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.

3. Pujari, A. (2008). *Data Mining Techniques*. 2nd edition. Universities Press.

d) BCA822D: Project Work/ Dissertation (Discipline Specific Elective)**Credit: 06****Total Marks: 100 Marks (Theory: 75 Marks, Internal Assessment: 25 Marks)****Workload: 4 Lecturers (Per Week), 4 Practicals (Per Week)****Course Objective**

The students will undergo one semester of project work based on the concepts studied in a subject of their choice. The objective is to train the students for the industry by exposing them to prototyping and development of real life software.

Course Learning Outcomes

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

1. develop a project plan based on informal description of the project.
2. implement the project as a team.
3. write a report on the project work carried out by the team and defend the work done by the team collectively.
4. present the work done by the team to the evaluation committee.

Detailed Syllabus**UNIT-I****15M**

The students will work on any project based on the concepts studied in core/elective/ skill based elective courses. Specifically, the project could be a research study, or a software development project.

UNIT-II**15M****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 who will be responsible for their lab classes.
- A maximum of four (04) projects would be assigned to one teacher.

UNIT-III**15M****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.
- 50 marks for continuous evaluation (to be awarded by the supervisor/mentor). Work carried out in each lab session will be assessed out of five marks (zero for being absent). Finally, the marks obtained will be scaled out of a maximum of 50 marks. For example, if

103

30 lab sessions are held in a semester, and a student has obtained an aggregate of 110 marks, then he/she will be assigned round $(110/(30*5))$ i.e. 37 marks.

- The students will submit only the soft copies of the report.
- The reports may be retained by the examiners.

Practical

Practical/discussion sessions based on the area of the project.

Skill-Enhancement Elective Course (SEC-I)

a) 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 Practicals (Per Week)

Course Objectives:

The objective of the course is to introduce the Mathematical Foundation which provides a strong foundation in the 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 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

15M

Algebra: Logarithms-Introduction, Definition, Laws of operations, change of base, Binomial theorem-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

15M

Matrix Algebra: Definition, types of matrices, algebra of matrices-addition of matrices, subtraction of matrices, multiplication of matrices, determinant of a matrix, Adjoint 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

15M

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 for n th term, sum of terms, Arithmetic mean, problems. Geometric

UNIT-IV

15M

Trigonometry: Introduction, Measurement of angles, trigonometric functions, relation between trigonometric functions, signs of trigonometric functions, trigonometric functions of standard angles.

[

UNIT-V

15M

Calculus: Limit of function, continuity of a function.

Differentiation: Derivative of a function of 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 V. K. 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 Mathematics-I, Rimple A, Pragatis Edition (IV)
4. B. S. Vatsa- Discrete Mathematics – New Age International Limited Publishers, New Delhi

b) 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)****Detailed Syllabus****Introduction to open office/MS office/Libreoffice(2L) WordProcessing: 20M**

FormattingText, Pages,Lists,Tables (5L)

Spreadsheets: 25M

Worksheets, Formattingdata, creatingcharts andgraphs, usingformulas and functions, macros, Pivot Table (6L)

PresentationTools: 20M

Addingand formattingtext, pictures,graphic objects, includingcharts, objects, formattingsslides, notes, hand-outs, slideshows, usingtransitions, animations (4L)

Books Recommended:

1. Sushila Madan ,Introduction to Essential tools,JBA,2009.
2. Anita Goel, ComputerFundamentals, Pearson,2012

ComputerLabBasedonOfficeAutomation: PracticalList forWORD:

1. Create a**telephonedirectory**.
 - Theheadingshould be16-pointArial Font in bold
 - Therest of the document should use10-pointfont size
 - Other headings shoulduse10-pointCourier New Font.
 - Thefooter should showthe pagenumberaswellas the date last updated.
2. Designatime-tableform foryour college.
 - Thefirst line should mention the name ofthe collegein 16-pointArial Font and should bebold.
 - Thesecond lineshouldgive thecoursename/teacher’s name and the departmentin 14- pointArial.
 - Leaveagap of 12-points.
 - Therest of the document should use10-pointTimes New Roman font.
 - Thefooter should containyour specifications as thedesignerand date of creation.
3. Create thefollowingonepagedocuments.
 - (a) Composeanote invitingfriends to aget-together atyour house, includingalist of things to bringwith them.
 - (b)Design acertificate inlandscapeorientation with a borderaroundthe document.
4. Create thefollowingdocument: A newsletterwith aheadlineand 2columns in portrait orientation, includingat least one imagesurrounded bytext.
5. Convert followingtext to atable, using commaas delimiter
Typefollowingas shown (do not bold).
Color, Style, Item Blue, A980, Van Red, X023, Car Green, YL724,Truck Name, Age, Sex Bob, 23, M Linda, 46, F Tom, 29, M
6. Prepare agrocerylist havingfour columns (Serial number, the name of theproduct, quantity and price) for themonthof April, 06.
 - Font specifications forTitle(Grocery List): 14-pointArial font in bold and italics.
 - Theheadings of thecolumns should bein 12-pointand bold.
 - Therest of the document should be in 10-pointTimes New Roman.
 - Leaveagap of 12-points after thetitle.
7. XYZPublications plans to releaseneew book designedasperyour syllabus. Design the

first page of the book as per the given specifications.

- (a) The title of the book should appear in bold using 20-point Arial font.
- (b) The name of the author and his qualifications should be in the center of the page in 16-point Arial font.
- (c) At the bottom of the document should be the name of the publisher and address in 16-point Times New Roman.
- (d) The details of the offices of the publisher (only location) should appear in the footer.

8. Create the following one-page documents. a)

Design a Garage Sales sign.

b) Make a sign outlining your rules for your bedroom at home, using a numbered list.

9. Enter the following data into a table given on the next page.

| Salesperson | Dolls | Trucks | Puzzles |
|-------------|-------|--------|---------|
| Amit | 1327 | 1423 | 1193 |
| Shivi | 1421 | 3863 | 2934 |
| Om | 5214 | 3247 | 5467 |
| Ananya | 2190 | 1278 | 1928 |
| Anupama | 1201 | 2528 | 1203 |
| Maharshi | 4098 | 3079 | 2067 |

Add a column Region (values: S, N, N, S, S, S) between the Salesperson and Dolls columns to the given table. Sort your table data by Region and within Region by Salesperson in ascending order:

Practical List for EXCEL

| | |
|------|---|
| Q1. | Create a student worksheet containing roll numbers, names and total marks. Open a document in Word and insert the excel worksheet using:- |
| i) | Copy/Paste |
| ii) | Embedding |
| iii) | Linking |
| Q2. | The term wise marks for APS class of 20 students are stored in 3 separate sheets named term1, term2 and term3. Create 4th worksheet that contains student names and their total and average marks for the entire year. Give proper headings using headers. Make |

the column headings bold and italic. The 4th worksheet should contain college name as the first line. Make it bold, italic and center it.

| | | | | | | |
|-----|---|----|----|---------|--------|----|
| Q3. | Using a simple pendulum, plot 1-T and 1-T ² graph. | | | | | |
| I | t1 | t2 | t3 | Mean(t) | T=t/20 | T2 |
| 70 | | | | | | |

| |
|----|
| 80 |
| 90 |

| | | | | | | | | |
|-------------------------------|---|-------|--------|---------|----|-------|-----|------|
| 100 | | | | | | | | |
| Q4. | Consider the following employee worksheet:- | | | | | | | |
| Full | Name | Grade | Basic | HRA | PF | Gross | Net | (VA) |
| (First | Last) | 1/2/3 | Salary | Vehicle | | | | |
| Allowance | | | | | | | | |
| HRA is calculated as follows: | | | | | | | | |
| Grade | HRA % (of Basic) | | | | | | | |
| 1 | 40% | | | | | | | |
| 2 | 35% | | | | | | | |
| 3 | 30% | | | | | | | |

Gross = Basic + HRA + VA

Net = Gross - PF

PF is 8% for all Grades

VA is 15000, 10000 and 7000 for Grades 1, 2 and 3.

i) Find max, min and average salary of employees in respective Grade

| | |
|------|--|
| ii) | Count no. of people where VA > HRA |
| iii) | Find out most frequently occurring grade. |
| iv) | Extract records where employee name starts with "A" has HRA > 10000 |
| v) | Print Grade wise report of all employees with subtotals of net salary and also grand totals. Use subtotal command. |
| vi) | Extract records where Grade is 1 or 2 and salary is between 10000 and 20000 both inclusive. |

Q5. In a meeting of a marketing department of an organization it has been decided that price of selling an item is fixed at Rs 40. It was resolved to increase the sale of more items and getting the profit of Rs 40,000/. Use Goal Seek to find out how many items you will have to sell to meet your profit figure.

| | |
|----|--|
| Q6 | To study the variation in volume with pressure for a sample of an air at constant temperature by plotting a graph for P-V and P-I/V. Sample observations are:- |
|----|--|

| Pressure (P) | Volume (V) | I/V | PV | P/V |
|--------------|------------|-----|----|-----|
| 75 | 20 | | | |
| 78.9 | 19 | | | |
| 83.3 | 18 | | | |
| 88.2 | 17 | | | |

Q7. Plot the chart for marks obtained by the students (out of 5) vs. frequency (total number of students in class is 50).

Q8. Create the following worksheet(s) containing a year wise sale figure of five salesmen in Rs.

| Salesman | 2002 | 2003 | 2004 |
|----------|-------|-------|--------|
| MOHAN | 10000 | 12000 | 20000 |
| MITRA | 15000 | 18000 | 50000 |
| SHIKHA | 20000 | 22000 | 70000 |
| ROHIT | 30000 | 30000 | 100000 |
| MANGLA | 40000 | 45000 | 125000 |

Apply the following Mathematical & Statistical functions:

i) Calculate the commission for each salesman under the condition :-

a) If total sales is greater than Rs. 3,00,000/-, then commission is 10% of total

sale made by the salesman.

b) Otherwise, 4% of total sale.

ii) Calculate the maximum sale made by each salesman.

iii) Calculate the maximum sale made in each year.

iv) Calculate the minimum sale made by each salesman. v)

Calculate the minimum sale made in each year.

vi) Count the no. of sales persons.

vii) Calculate the cube of sales made by Mohan in the year 2002.

viii) Find the difference in sales by salesman Mitra between the year 2002 and 2003.

Find the absolute value of difference.

ix) Also calculate the Mode, Stddev, Variance, Median for the sale made by each salesman.

ix) Calculate the year wise Correlation coefficient between the sales man Mohan and Mitra year wise

| | | | |
|----------|--|-------|--------|
| Q9. | The following table gives an year wise sale figure of five salesmen in Rs. | | |
| Salesman | 2000 | 2001 | 2002 |
| S1 | 10000 | 12000 | 20000 |
| S2 | 15000 | 18000 | 50000 |
| S3 | 20000 | 22000 | 70000 |
| S4 | 30000 | 30000 | 100000 |
| S5 | 40000 | 45000 | 125000 |

v) Calculate total sale year wise.

vi) Calculate the net sales made by each salesman

vii) Calculate the commission for each salesman under the condition :-

c) If total sales is greater than Rs. 4, 00,000/-, then commission is 5% of total sale made by the salesman.

d) Otherwise, 2% of total sale.

viii) Calculate the maximum sale made by each salesman.

ix) Calculate the maximum sale made in each year.

x) Draw a bar graph representing the sale made by each salesman.

xi) Draw a pie graph representing the sale made by salesmen in year 2001.

| | |
|------|--|
| Q10. | Consider the following worksheet for APS 1st year students:- |
|------|--|

| S.No. | Name | PH | CH | BY | MT | CS | Total | % | Grade |
|----------------------------------|---------|----|----|----|----|----|-------|---|-------|
| Marks | | | | | | | | | |
| 1 | | | | | | | | | |
| 2 | | | | | | | | | |
| Grade is calculated as follows:- | | | | | | | | | |
| If % ≥ 90 | Grade A | | | | | | | | |
| If % ≥ 80 & < 90 | Grade B | | | | | | | | |
| If % ≥ 70 & < 80 | Grade C | | | | | | | | |
| If % ≥ 60 & < 70 | Grade D | | | | | | | | |

Otherwise students will be declared fail.

i) Calculate Grade using if function

ii) Sort the data according to total marks

- iii) Apply filter to display the marks of the students having more than 65% marks.
- iv) Draw a pie chart showing % marks scored in each subject by the topper of the class. v) Draw the doughnut chart of the data as in (iv)
- vi) Enter the S.No. of a student and find out the Grade of the student using VLOOKUP.
- vii) Extract all records where name
 - a) Begins with "A"
 - b) Contains "A"
 - c) Ends with "A"

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.

c) 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, Lab Demonstrators.

Detailed Syllabus**UNIT-I****15M**

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**15M 1**

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**15M 1**

Testing & troubleshooting of personal computer problems Interfacing primary Memory card in the CPU. Interfacing Harddisk 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**15M**

Installation of Operating System like Windows7, Windows8, Windows 2008, various Linux flavors like Ubuntu, SUSE, RedHat, 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**10M**

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

UNIT-VI**10M**

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

Reference

1. Adane Nega Tarekegn, Alemu Kumilachew Tegegne, "A simple guide to Computer Maintenance and Troubleshooting", Lambert Publication.
2. [Craig Zacker](#), [John Rourke](#), "Pc Hardware, A complete reference", McGraw Hill
3. Manahar Lotia, "Modern Computer Hardware Course", BPB Publication

Skill-Enhancement Elective Course (SEC-2)

a) 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 interprocess 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

25M

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, ComparingFiles.

UNIT-II

15M

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

15M

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**25M**

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

Books recommended.

Text Books

1. M.G.Venkateshmurthy: Introduction to Unix & Shell Programming, PearsonEducation.

References

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

b) BCA224B: Object Oriented Programming Using C++() Sec-2**Credit: 06****Total Marks: 100 Marks(Theory: 75 Marks, Internal Assessment: 25 Marks)****Workload: 4 Lecturers (Per Week), 4 Practicals (Per Week)****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 be 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**15M****Introduction to C and C++**

History of C and C++, Overview of Procedural Programming and Object-Oriented Programming, Using main() function, Compiling and Executing Simple Programs in C++.

UNIT 2:**15M****Classes and Objects:**

Encapsulation, information hiding, abstract data types, Object & classes, attributes, methods, C++ class declaration, State identity and behavior of an object, Constructors and destructors, instantiation of objects, Default parameter value, object types, C++ garbage collection, dynamic memory allocation, Metaclass/abstract classes.

UNIT 3:**15M****Inheritance and Polymorphism:**

Inheritance, Class hierarchy, derivation – public, private & protected, Aggregation, composition vs classification hierarchies, Polymorphism, Categorization of polymorphism techniques, Method polymorphism, Polymorphism by parameter, Operator overloading, Parametric polymorphism,

Conditional Statements and Iterative Statements

Conditional Statements (if construct, switch- case construct), Understanding syntax and utility of Iterative Statements (while, do-while, and for loops), Use of break and continue in Loops, Using Nested Statements (Conditional as well as Iterative)

UNIT 4:
User Define, Functions and Arrays

15M

The Need of Functions, User Defined and Library Function, Prototype of Functions, Prototype of main() Function, Calling of Functions, Function Arguments, Argument Passing: Call By Value and Call By Reference, Return Values. Nesting of Function, Recursion, Array as Function Argument, Command Line Arguments, Basics of Pointers, Pointers Operators, Pointer Arithmetic, Pointers and Function, Pointer and Strings. Preprocessor and its Advantages.

UNIT 5:

Structure and Pointers

15M

Storage Class Specifier- Auto, Extern, Static, Register. Defining Structure, Declaration of Structure Variable, Type def, Accessing Structure Members, Member Access Operator, Nested Structures, Array of Structure, Structure Assignment, Structure as Function Argument, Function that Return Structure, Union. Understanding a Pointer Variable, Pointer to Structure, Pointers within Structure, Introduction to Static and Dynamic Memory Allocation, The Process of Dynamic Memory Allocation, DMA Functions : malloc(), calloc(), free(), realloc(), sizeof() Operator. C++ Classes and Object.

File/O

Opening and closing a file (use ofstream 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. E Balaguruswamy, "Object Oriented Programming with C++", Tata McGraw-Hill Education, 2008.
2. Object Oriented Programming in TURBO C++ ,Galgotia, Robert Lafore
3. 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. BjarneStroustrup, "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 usingC++Lab

1. WAPto print thesumand product ofdigits ofan integer.
2. WAPto reverseanumber.
3. WAPto computethesum ofthe first n terms ofthe followingseries $S=1+1/2+1/3+1/4+.....$
4. WAPto computethesum ofthe first n terms ofthe followingseries $S=1-2+3-4+5.....$
5. Writea function that checks whetheragiven stringis Palindromeornot. Usethis function to find whetherthestring entered byuseris Palindromeornot.
6. Writea function to find whetheragiven no. is primeornot. Usethesameto generate theprimenumbers less than 100.
7. WAPto computethe factors ofagiven number.
8. Writeamacro that swaps two numbers. WAPto useit.

9. WAP to print a triangle of stars as follows (take number of lines from user):

```

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

```

10. WAP to perform following actions on an array entered by the user:

- i) Print the even-valued elements
- ii) Print the odd-valued elements
- iii) Calculate and print the sum and average of the elements of array
- iv) Print the maximum and minimum element of array
- v) Remove the duplicates from the array
- vi) Print the array in reverse order

The program should present a menu to the user and ask for one of the options. The menu should also include options to re-enter array and to quit the program.

11. WAP that prints a table indicating the number of occurrences of each alphabet in the text entered as command line arguments.

12. Write a program that swaps two numbers using pointers.

13. Write a program in which a function is passed address of two variables and then alters its contents.

14. 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.

15. 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.

16. Write a menu driven program to perform following operations on strings:

- a) Show address of each character in string
- b) Concatenate two strings without using strcat function.
- c) Concatenate two strings using strcat function.
- d) Compare two strings
- e) Calculate length of the string (use pointers)
- f) Convert all lowercase characters to uppercase
- g) Convert all uppercase characters to lowercase
- h) Calculate number of vowels
- i) Reverse the string

17. Given two ordered arrays of integers, write a program to merge the two arrays together into an ordered array.

18. WAP to display Fibonacci series (i) using recursion, (ii) using iteration

19. WAP to calculate Factorial of a number (i) using recursion, (ii) using iteration

20. Write a program to calculate GCD of two numbers (i) with recursion (ii) without recursion.
21. Create a Matrix class using templates. Write a menu-driven program to perform following Matrix operations (2-D array implementation):
 - a) Sum
 - b) Difference
 - c) Product
 - d) Transpose
22. 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).
23. Create a class Triangle. Include overloaded functions for calculating area. Overload assignment operator and equality operator.
24. Create a class Box containing length, breadth and height. Include following methods in it:
 - a) Calculate surface Area
 - b) Calculate Volume
 - c) Increment, Overload ++ operator (both prefix & postfix)
 - d) Decrement, Overload -- operator (both prefix & postfix)
 - e) Overload operator == (to check equality of two boxes), as a friend function
 - f) Overload Assignment operator
 - g) Check if it is a Cube or cuboid
 Write a program which takes input from the user for length, breadth and height to test the above class.
25. Create a structure Student containing fields for Roll No., Name, Class, Year and Total Marks. Create 10 students and store them in a file.
26. Write a program to retrieve the student information from file created in previous question and print it in following format:


```
Roll No.   Name   Marks
```
27. Copy the contents of one text file to another file, after removing all whitespaces.
28. Write a function that reverses the elements of an array in place. The function must accept only one pointer value and return void.
29. 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.

C) BCA224C: Introduction to Multimedia() SEC-2**Credit: 06****Total Marks: 100 Marks(Theory: 75 Marks, Internal Assessment: 25 Marks)****Workload: 4 Lecturers (Per Week), 4 Practicals (Per Week)****CourseObjective**

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****10M****Introduction to Multimedia**

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

UNIT-II**10M****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**10M****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**15M****Image fundamentals and representations**

ColourScience ,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**15M****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**15M****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.

Other Discipline:**Credit: 06 each)-****GEC-1 to GEC-6****Total Marks: 100 Marks(Theory: 75 Marks, Internal Assessment: 25 Marks)****Workload: 4 Lecturers (Per Week), 4 Practicals (Per Week)****ORGANIZATIONAL BEHAVIOUR (GENERIC ELECTIVE COURSE)****Course Objective:**

1. To help the students to develop cognizance of the importance of human behaviour.
2. To enable students to describe how people behave under different conditions and understand why people behave as they do
3. To provide the students to analyse specific strategic human resources demands for future action.
4. To enable students to synthesize related information and evaluate options for the most logical and optimal solution such that they would be able to predict and control human behaviour and improve results.

Course Outcomes:

On completion of this course, the students will be able to

CO1: Demonstrate the applicability of the concept of organizational behaviour to understand the behaviour of people in the organization.

CO2: Demonstrate the applicability of analyzing the complexities associated with management of individual behaviour in the organization.

CO3: Analyze the complexities associated with management of the group behaviour in the organization. CO4: Demonstrate how the organizational behaviour can integrate in understanding the motivation (why) behind behaviour of people in the organization.

Deatiled Syllabus**UNIT-I****10M**

Meaning, Fundamental concepts, Definition, Approaches to OB, Characteristics and limitations of OB, Challenges and Opportunities of OB, Models of OB.

UNIT-II**15M**

Personality: Definition, Features, Big five model, MBTI, Johari Window, Managerial Implications of Personality. Perceptions and Attributions: Definition Features, factors affecting perception, Process. Attribution, perceptual and attribution errors, Managerial Implications of Perception.

UNIT-III**15M**

Learning: Definition, Features, Classical and operant conditioning, social learning theory, Behavioral modification. Attitude: Definition, Features, ABC model of Attitude, Managerial Implications of Attitude.

UNIT-IV**10M**

Motivation: Concept, Definition, Features, Types of Motivation, Process, Managerial Implications of Motivation. Leadership: Concept, Definition, Leadership Styles, Transactional and Transformational Leadership, Leadership development.

UNIT-V**15M**

Groups and Teams: Definition, Features, Group development stages, Group vs. Teams, Managing and developing effective teams. Conflict Management: Definition, Features, Types of Conflict, Conflict Resolution Strategies, Relationship between Conflict and Performance.

UNIT-VI**15M**

Organizational Culture: Elements and dimensions of organizational culture, Importance of organizational culture in shaping the behaviour of people. Organizational Change: Understanding the issues and managing change, Approaches to organizational change.

Text Books

Robbins, S. P., & Judge, T. (2013). *Organizational behavior* (15th ed.). Boston: Pearson. Newstrom J. W., & Davis, K. (2011). *Human behavior at work* (12th ed.). Tata McGraw Hill Nelson, D, Quick, J.C., & Khandelwal, P., (2011). *ORGB* . Cengage Learning.

Reference Books

Pareek. U. (2010). *Understanding Organizational Behavior* (2nd ed.). Oxford University Press
Schermerhorn, J. R., Osborn, R.N., Hunt, M.U.J (2016). *Organizational Behavior* (12th ed.). Wiley.

Environmental Science () Ability Enhancement Compulsory-(AECC)**Credit:04****Total Marks: 100 Marks(Theory: 75 Marks, Internal Assessment: 25 Marks)****Workload: 4 Lecturers (Per Week), 4 Practicals (Per Week)****CourseObjective**

The Environmental Studies major prepares students for careers as leaders in understanding and addressing complex environmental issues from a problem-oriented, interdisciplinary perspective.

CourseLearning Outcomes

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

1. Understand the concept of environmental studies, Ecology, food chain, food webs and ecological pyramids, Ecosystem and how it functions.
2. Analyse the concept of biodiversity and its conservation approaches, its value and threat.
3. Familiarize with Natural resources, its importance and conservation, and also the role an individual play in its conservation.
4. Explain the impact of industrialization to the environment and the different types of environmental pollution.

UNIT-I**25M**

Environmental studies – meaning – scope- importance, Ecology and ecosystem, biodiversity and its conservation, and natural resources – meaning of ecology – structure and function of and ecosystem- producers–consumers–decomposers–energyflowintheecosystem–ecologicalsuccession–foodchain– food webs and ecological pyramids . Ecosystem – concept – types of ecosystems – structure and functions of forest ecosystem, grassland ecosystem – desert ecosystem, aquatic ecosystem.

UNIT-II**25M**

Biodiversityanditsconservation-introduction–definition– genetic,speciesandecosystemdiversity–value of biodiversity – biodiversity at global, natural and local levels – India as a mega – diversity nation – hot – spots of diversity – threats to diversity – conservation of diversity in in-situ, ex-situ. Natural resources – features– airresources,forestresources,waterresources,mineralresources–consequences–conservationof natural resources – role of an individual in conservation of naturalresources.

UNIT-III**15M**

Industry and environment – pollution – environmental pollution – soil pollution – air pollution – water pollution – thermal pollution – noise pollution – causes, effects and control measures – waste management – waste minimisation through cleaner technologies – reuse and recycling – solid waste management. Environment ethics – waste land reclamation – consumerism and waste products.

UNIT-IV**15M**

Social issues and environment – unsustainable to sustainable development – urban problems related to energy – water conservation – water harvesting – resettlement and rehabilitation of people. Human population and environment–populationgrowth–variationamongnations– populationexplosion–environmentandhuman health – human rights – value education – women and childwelfare.

Reference Books

1. Misra S.P. and Pandey S.N., '*Essential environmental studies*, 'Anebooks India, New Delhi.
2. Kiran B. Chokkas and others, '*Understanding environment*, ' Sage Publications, New Delhi
3. Arumugam N. and Kumaresan V., '*Environmental Studies*, 'Saras Publications Kanyakumari
4. Benny Joseph, '*Environmental Studies*, 'Tata McGraw-Hill Publishing Co. Ltd., New Delhi.
5. Anjaneyalu, Y. 2004, Introduction to Environmental Science. BS Publication, Hyderabad, A.P. India