The University of Burdwan

Syllabus of 4-Year Honours in Computer Applications

Under Curriculum and Credit Framework for Undergraduate Programme (CCFUP) as per NEP, 2020 w.e.f. 2023-2024

Preamble

The new curriculum of the four-year undergraduate program under NEP, for Computer Application aims to develop the core competence in computing and problem solving amongst its graduates. Informally, "Learning to learn" has been the motto of the department since its inception. The curriculum thus focuses on building theoretical foundations in Computer Application to enable its pupils to think critically when challenged with totally different and new problems. It imbibes the following **Student-Centric** features of NEP2020:

Flexibility to Exit: In order to support early exits, the curriculum aims to develop employability skills early. This has been done so that the outcomes of the 4 yr degree is not compromised as we believe that all but a few students will go for the full 4-year degree. As programming is at the heart of computing it is proposed to have two programming courses early so that the students can develop good programming skills in the first year. At the same time students are familiarized with the hardware of computers early on.

Employability: Industry demand in the IT sector has changed considerably in the past few years. With the humongous amount of data coming from all the domains like medical data, social networking data, astronomical data, education, etc., automating information extraction and analysis of data is the only way forward to leverage the available data for the future. The curriculum aims to equip the students with tools and techniques of Artificial Intelligence, Machine Learning and a pathway on Data Science if the student so desires. Having said this, there is no replacement for the foundational courses like programming, data structures and algorithms. With two courses on programming and three courses on data structures and algorithms together, a strong foundation will be laid down for problem solving.

Multidisciplinary/Minor: The curriculum provides two pathways one of Computer Application minor and the other of interdisciplinary, to the students from other disciplines. Those who want to earn a minor in CS will be required to choose the first pathway whereas those who simply want to apply IT in the domain of their interest can choose the second path way.

Research: With the option to obtain specialization in an area of their choice, the curriculum prepares the students to take up research projects in their final year.

Semester	Course Type & Course Code	Name of the Course	Credit	Lect.	Tuto.	Pract./ Viva	Full Marks	Distribution of Marks		
								Theory	Pract. / Tuto./ Viva-voce	Internal Assessment
I	**Major/DS Course (Core)** **BCA 1011**	Computer Fundamentals and Introduction to C Programming	**4**	**3**	**0**	**1**	**75**	**40**	**20**	**15**
	Minor Course **BCA 1021**	Mathematics-I (**Only for BCA Students)**	**4**	**3**	**0**	**1**	**75**	**60**	**NIL**	**15**
	Multi/Interdisciplinary **BCA 1031**	Basic IT tools **(For other discipline)**	**3**	**2**	**0**	**1**	**50**	**40**	**NIL**	**10**
	Ability Enhancement Course [L_1-1 MIL] **….. 1041**	**Arabic/ Bengali/ Hindi/ Sanskrit/ Santali/ Urdu** **or Equvlnt. Course from SWAYAM or UGC recognized others**	**2**	**2**	**0**	**0**	**50**	**40**	**0**	**10**
	Skill Enhancement Course (SEC) **BCA 1051**	PC Software	**3**	**0**	**0**	**3**	**50**	**NIL**	**40**	**10**
	Common Value Added (CVA) Course **CVA1061**	Environmental Science/ Education	**4**	**3**	**0**	**1**	**100**	**60**	**20**	**20**
	Total		**20**				**400**			

Semester	Course Type & Course Code	Name of the Course	Credit	Lect.	Tuto.	Pract. /Viva	Full Marks	Distribution of Marks		
								Theory	Pract. / Tuto./ Viva-voce	Internal Assessment
II	**Major/DS Course (Core) BCA 2011**	Data Structures	**4**	**3**	**0**	**1**	**75**	**40**	**20**	**15**
	Minor Course BCA 2021	Mathematics-II (**Only for BCA Students)**	**4**	**3**	**0**	**1**	**75**	**60**	**NIL**	**15**
	Multi/Interdisciplinary BCA 2031	Introduction to Internet **(For other discipline)**	**3**	**2**	**0**	**1**	**50**	**40**	**NIL**	**10**
	Ability Enhancement Course (AEC)[L_2-1] ENGL 2041	Functional English or Equvlnt. Course from SWAYAM or UGC recognized others	**2**	**2**	**0**	**0**	**50**	**40**	**0**	**10**
	Skill Enhancement Course (SEC) BCA 2051	Python Programming	**3**	**0**	**0**	**3**	**50**	**NIL**	**40**	**10**
	Common Value Added (CVA) Course CVA 2061	Understanding India/ Digital & Technological Solutions / Health & Wellness, Yoga Education, Sports and Fitness	**4**	**3**	**1/0**	**0/1**	**100**	**80/60**	**0/20**	**20**
	Total		**20**				**400**			

Semester	Course Type	Name of the Course	Credit	Lect.	Tuto	Pract /Viva	Full Marks	Distribution of Marks		
								Theory	Pract/ tuto./ Viva-voce	Internal Assessment
III	**Major/DS course (Core) BCA 3011**	Object Oriented Programming Using JAVA	5	4	0	1	75	40	20	15
	Major/DS course (Core) BCA 3012	Operating System	5	4	0	1	75	40	20	15
	Minor Course (Vocational Education. & Training.) **MSR 3021** **OR** **HRM 3021** **OR** **RSA 3021**	(Intermediate Level Course) Medical Sales Representative OR Human Resource Management OR Retail Sales Associate	4	3	1	0	75	60	0	15
	Multi/ Interdisciplinary BCA 3031	Introduction to Cyber Security **(For other discipline)**	3	3	0	0	50	40	0	10
	Ability Enhancement Course (AEC) [L_1-2 MIL] ... 3041	[Arabic/ Bengali/Hindi/ Sanskrit/ Santhali/Urdu] or Equvlnt. Course from SWAYAM or UGC recognized other platforms	2	2	0	0	50	40	0	10
	Skill Enhancement Course (SEC) **BCA 3051**	Unix and Shell Programming	3	0	0	3	50	NIL	40	10
	Total		22				375			

Semester	Course Type	Name of the Course	Credit	Lect.	Tuto.	Pract /Viva	Full Marks	Distribution of Marks		
								Theory	Pract/tuto./Viva-voce	Internal Assessment
IV	**Major/DS course (Core) BCA 4011**	Digital Logic	5	4	0	1	75	40	20	15
	Major/DS course (Core) BCA 4012	Design and Analysis of Algorithms	5	4	0	1	75	40	20	15
	Major/DS course (Core) BCA 4013	Software Engineering	5	4	0	1	75	40	20	15
	Minor Course BCA 4021	Discrete Structure **(Only for BCA Students)**	4	3	1	0	75	60	0	15
	Minor Course BCA 4022	Principles of Management **(Only for BCA Students)**	4	3	1	0	75	60	0	15
	Ability Enhancement Course (AEC) [L_2-2] ENGL 4041	Language and creativity or Equvlnt. Course from SWAYAM or any other UGC recognized platform	2	2	0	0	50	40	0	10
	Total		25				425			

Semester	Course Type & Couse Code	Name of the Course	Credit	Lect.	Tuto.	Pract. /Viva	Full Marks	Distribution of Marks		
								Theory	Pract. / Viva-voce	Internal Assessment
V	**Major/DS Course (Core) BCA 5011**	Database Management System	5	4	0	1	75	40	20	15
	Major/DS Course (Core) BCA 5012	Theory of Computation	5	4	1	0	75	60	0	15
	Major/DS Course (Core) BCA 5013	PHP Programming	5	4	0	1	75	40	20	15
	Minor Course (Vocational Education. & Training.) **MSR 5021 OR HRM 5021 OR RSA 5021**	(Intermediate Level Course) Medical Sales Representative OR Human Resource Management OR Retail Sales Associate	4	3	1	0	75	60	0	15
	Internship (for all students) INT 5081	Internship/Project and Viva Voce	2	0	0	2	50	00-50-00 (Project - 30 + Viva - 20)		
	Total		**21**				**350**			

Semester	Course Type & Couse Code	Name of the Course	Credit	Lect.	Tuto.	Pract. /Viva	Full Marks	Distribution of Marks		
								Theory	Pract. / Viva-voce	Internal Assessment
VI	**Major/DS Course (Core) BCA 6011**	Data Communication and Networking	4	3	0	1	75	40	20	15
	Major/DS Course (Core) BCA 6012	Artificial Intelligence	4	3	0	1	75	40	20	15
	Major/DS Course (Core) BCA 6013	Microprocessor	4	3	0	1	75	40	20	15
	Major/ DS Course (Core) BCA 6014	Internet & E-Commerce	4	3	0	1	75	40	20	15
	Minor Course (Vocational Education. & Training.) **MSR 6021 OR HRM 6021 OR RSA 6021**	(Intermediate Level Course) Medical Sales Representative OR Human Resource Management OR Retail Sales Associate	4	3	1	0	75	60	0	15
	Total		**20**				**375**			

Semester	Course Type& Couse Code	Nameofthe Course	Credit	Lect.	Tuto.	Pract. /Viva	Full Marks	Distribution of Marks		
								Theory	Pract./ Tuto/ Viva-Voce	Internal Assessment
VII	**Major/DS Course (Core) BCA7011**	Internet Technology	6	4	0	2	75	40	20	15
	Major/DS Course BCA7012	Compiler Design	6	4	0	2	75	40	20	15
	Major/DS Course BCA7013	Advanced Data Structure &Algorithm	6	4	0	2	75	40	20	15
	Major/DS Course BCA7014	Soft Computing	6	4	0	2	75	40	20	15
	Minor Course BCA7021	Management Process and Organizational Behaviour / MOOCS Course* **(For BCA Course only)**	4	3	1	0	75	60	0	15
	Total		**28**				**375**			

*At least 4 Credit or 16 weeks course from MOOCS platform like SWAYAM , NPTEL , Udeamy , Coursera, edX etc. on relevant discipline can be taken subject to the approval of concerned academic department.

Semester	CourseType& Couse Code	Nameofthe Course	Credit	Lect.	Tuto.	Pract. /Viva	Full Marks	Distribution of Marks		
								Theory	Pract./ Tuto/ Viva-Voce	Internal Assessment
VIII For UG Hons. With Research Project/ Dissertation	**Major/DS Course (Core) BCA8011**	Network Security &Cryptography	6	4	0	2	75	40	20	15
	Minor Course BCA 8021	Operations Research / MOOCS Course* (**Only for BCA students**)	4	3	1	0	75	60	0	15
	Research Project BCA8091	Research Project/ Dissertation	12	0	0	12	225	Seminar Presentation, Preparation & Submission of Research Project/Dissertation-135+ Viva–90		
VIII ForUG Hons. Without Research Project/ Dissertation	**Major/DS Course(Core) BCA 8011**	Network Security &Cryptography	6	4	0	2	75	40	20	15
	Major/DS Course(Core) BCA 8012	Computer Graphics	4	3	0	1	75	40	20	15
	Major/DS Course (Core) BCA8013	Machine Learning	4	3	0	1	75	40	20	15
	Major/DS Course(Core) BCA 8014	System Programming Using UNIX/Linux	4	3	0	1	75	40	20	15
	Minor Course BCA 8021	Operations Research (Only for BCA students)	4	3	1	0	75	60	0	15
	Total		**22**				**375**			

*At least 4 Credit or 16 weeks course from MOOCS platform like SWAYAM , NPTEL , Udeamy , Coursera, edX etc. on relevant discipline can be taken subject to the approval of concerned academic department.

Honours students not undertaking research will do 3 courses for 12 credits in lieu of a research project / Dissertation.

Students will be awarded UG Degree BCA (Honours/ Honours with Research) in the relevant Discipline / Subject provided they secure 178 credits.

Students who secure a minimum of 7.5 CGPA in the first six semesters and wish to undertake research at the undergraduate level can choose Honours with research degree in the fourth year. They should do a research project or dissertation under the guidance of a faculty member of the concerned Department of the University/ Institute. The research project/dissertation will be in the major discipline, Computer Application. The students who secure 12 credits from a research project/dissertation, will be awarded UG Degree in BCA(Honours with Research).

SEMESTER – I

BCA 1011: Computer Fundamentals and Introduction to C Programming

Course Learning Outcomes: After successful completion of the Course a student will be able to:

- **Learn about basic Components of a Computer.**
- **Develop problem solving skills coupled with top-down design principles.**
- **Become skilled at developing simple algorithms and flow charts.**
- **Convert the algorithms into simple C programs.**
- **Develop simple C programs for solving real life problems.**

Credit :3 45 Hours

Introduction – Introduction to computers – Evolution – Generation of Computers – Computers Hierarchy – Applications of Computers – Number System – Binary, Hexa, Octal, BCD System - Boolean Algebra – laws – logic gates – simplification of Boolean expression – k-map – sum of products – product of sums. 15**hours**

Introduction to computers and operating environment, program development cycle, Algorithm - Representation of Algorithms, Pseudocode, Flowchart & Decision tables, Structured Programming and Modular Programming, Introduction to C. 10**hours**

Data Types and sizes, variable declaration, operators, type conversion, conditional expressions, special operators, precedence rule. Control Structures- statements and blocks, if, switch, while, for, do-while, break, continue, goto and labels. [20%] 10**hours**

Functions & Program structure, recursion, arrays and pointers, structures and unions, standard I/O,formatted I/O, standard library functions. Files handling and pre-processing. String processing in C**.** **10hours**

Books:

1. Programming in C-B.S. Gottfried (Sahaum Series)
2. Programming in ANSI C- E. Balaguruswami(TMH)

BCA 1011: Computer Fundamental and Introduction to C Programming (Practical)

Credit :1 30 Hours

1. WAP to print the sum and product of digits of an integer.

2. WAP to reverse a number.

3. WAP to compute the sum of the first n terms of the following series S = 1+1/2+1/3+1/4+……

4. WAP to compute the sum of the first n terms of the following series S =1-2+3-4+5…….

5. Write a function that checks whether a given string is Palindrome or not. Use this function to find whether the string entered by user is Palindrome or not.

6. Write a function to find whether a given no. is prime or not. Use the same to generate the prime numbers less than 100.

7. WAP to compute the factors of a given number.

8. Write a macro that swaps two numbers. WAP to use it.

9. 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 alter 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 to get 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. WAP to calculate GCD of two numbers (i) with recursion (ii) without recursion.

21. Create 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

BCA 1021: Mathematics-I

This course introduces the students to the fundamental concepts and topics of linear algebra and vector calculus, whose knowledge is important in other computer courses. The course aims to build the foundation for some of the core courses in later semesters.

Course Learning Outcomes After successful completion of this course, the student will be able to:

- **Perform operations on matrices and sparse matrices**
- **Compute the determinant, rank and eigenvalues of a matrix**
- **Perform operations on vectors, the dot product and cross product**
- **Represent vectors geometrically and calculate the gradient, divergence, curl**
- **Apply linear algebra and vector calculus to solve problems in sub-disciplines of computer science**

Credit :4 60 Hours

Algebra: Sets, Union and Intersection, Complement, Mapping, Composition, notion of a Group, Ring , Field with simple examples. **10hours**

Complex Number: Modulus and amplitude, De Moiver's theorem **05hours**

Polynomials, Division algorithm, Fundamental theorem of classical algebra (Proof not required),Descartes rule of sign and their application, Relation between roots and coefficients; symmetric function of roots, Transformation of polynomial equation, Cardon's solution of cubic equation, Determinants, Addition and Multiplication of Matrices, Inverse of a Matrix ; Solution of linear equations in three variables by Cramer's rule and solution of three line linear equations by matrix inversion methods**.** **20hours**

Vector spaces, Subspaces, Bases and Dimensions, Co-ordinates, Linear Transformation, The Algebra of Linear Transformations. Vector Algebra: Scalars & vectors, vector addition, linear combination of vectors, condition of colinearity of three points, scalar and vector products, scalar triple product and vector triple product. **15hours**

Analytical Geometry: Translation and rotation of rectangular axes, invariants, general equation of second degree-reduction to standard forms and classification. Plane polar equation of a straight line, circle, ellipse, parabola and hyperbola. **10hours**

Books:

1. A Text book of Algebra- B.K. Lahiri & K. C. Roy

2. Linear Algebra- Das & Roy

3. Co-ordinate Geometry- S. L. Loney

4. Differential Calculus- Das and Mukherjee

5. Integral Calculus - Das and Mukherjee

Multi/Interdisciplinary courses

(For other discipline)

Semester-I

BCA 1031: Basic IT Tools(Theory)

Credit: 03 **45 Hours**

The goal of this course is to present overview of IT tools used in day-to-day use of computers and data base operations. The Course has been designed to provide knowledge on various hardware and software components of computer, operating system, various packages used for different applications, data base concepts & operations and various issues related to IT and application of IT.

Course Learning Outcomes:

On successful completion of the Course, a student will:

i. Acquire the foundation level knowledge required to understand computer and its operations.
ii. Understand the hardware and software components of the computer.
iii. Understand the basic concept of operating system and get knowledge about various different operating systems.
iv. Understand to use the packages of word processing, spread sheet and presentation in detail.
v. Understand various data base concepts and operations.
vi. Understand the issues related to IT and IT applications
vii. Prepare research and academic related presentations.

Syllabus

Introduction – Introduction to computers – Evolution – Generation of Computers – Computers Hierarchy – Applications of Computers. (5 Hrs)

Windows Basics – Introduction to word – Editing a document - Move and Copy text - Formatting text & Paragraph
– Enhancing document – Columns, Tables and Other features. (10 Hrs)

Introduction to worksheet and shell – getting started with Excel – Editing cell & using Commands and functions – Moving & Copying, Inserting & Deleting Rows & Columns - Printing work sheet. (5 Hrs)

Creating charts – Naming ranges and using statistical, math and financial functions, database in a worksheet – Additional formatting commands and drawing toolbar – other commands & functions – multiple worksheet and macros. (10 Hrs)

Introduction to Database Development: Database Terminology, Objects, Creating Tables, working with fields, understanding Data types, changing table design, Assigning Field Properties, Setting Primary Keys, select data with queries: Creating simple Query by design & by wizard (10 Hrs)

Overview of Power point – presenting shows for corporate and commercial using Power point –Introduction to Desktop publishing – Computer viruses – Introduction to Internet – Web features. (5 hrs)

Reference Books:

i. Swinford, E., Dodge, M., Couch, A., Melton, B. A. (2013). Microsoft OfficeProfessional 2013. United States: O'Reilly Media.

ii. Wang, W. (2018). Office 2019 For Dummies. United States: Wiley. Microsoft

iii. Lambert, J. (2019). Microsoft Word 2019 Step by Step. United States: Pearson Education.

iv. Jelen, B. (2013). Excel 2013 Charts and Graphs. United Kingdom: Que.

v. Alexander, M., Jelen, B. (2013). Excel 2013 Pivot Table Data Crunching. UnitedKingdom: Pearson Education.

vi. Alexander, M., Kusleika, R. (2018). Access 2019 Bible. United Kingdom: Wiley.

BCA 1051: PC Software

Course Learning Outcomes: After successful completion of the Course:

- **Learners will be able to claim proficiency in MS-Office.**
- **Learners will able to independently create professional-looking documents and presentations.**
- **Learners will be familiar with some advanced Word Power Point and Excel functions.**
- **Learners will be able to prepare research and academic related presentations.**

Credit :3 45 Hours

DOS: Introduction to DOS, internal and external commands, batch files (Autoexec.bat, config.sys), Line editors.

MS Windows: Desk top cell user interface action, icon on desktop, closing windows, renaming icons, resizing windows (maximizing and minimizing), control panel.

MS Word: Overview, creating, saving, opening, importing, exporting, and inserting files, formatting pages, paragraphs and sections, indents and outdents, creating lists and numbering. Headings, styles, fonts and font size, editing, positioning, viewing texts, searching and replacing text, inserting page breaks, page numbers, bookmarks, symbols, and dates. Using tabs and tables, header, footer, and printing,

MS Excel: Worksheet overview, entering information, worksheet creation, opening and saving workbook, formatting numbers and texts, protecting cells, producing charts, and printing operations.

MS Access: Introduction, understanding databases, creating tables, queries, forms, reports, adding graphs to your reports. PowerPoint: Slide creation with PowerPoint.

BOOKS:

1. Computer Fundamentals – Raja Raman – Prentice Hall of India .

2. Digital Circuits & Design – S.Salivahanan, S.Arivazhagan – Vikas Publishing

House Pvt Ltd.

3. Digital logic & Computer Design- M.Mano- Prentice Hall of India.

SEMESTER – II

BCA 2011: Data Structures

Course Learning Outcomes: (After the completion of course, the students will have ability to):

- **To be familiar with fundamental data structures and with the manner in which these data structures can best be implemented; become accustomed to the description of algorithms in both functional and procedural styles**
- **To have knowledge of complexity of basic operations like insert, delete, search on these data structures.**
- **Ability to choose a data structure to suitably model any data used in computer applications.**
- **Design programs using various data structures Binary and general search trees, heaps etc.**
- **Ability to assess efficiency tradeoffs among different data structure implementations.**

Implement and know the applications of algorithms for sorting, searching etc.

Credit :3 45Hours

Introduction: Introduction to algorithm, analysis for space and time requirements. **5hours**

Linear data structures and their sequential representation: Array, Stack, queue, circular queue, dequeue and their operation's and applications**.** **10hours**

Linear data structures and their linked representation: linear linked list, doubly linked list, linked stack and linked queue and their operation's and applications. **10hours**

Nonlinear data structure: Binary trees, binary search trees, representations and operations. Thread representations, sequential representations, graphs, and their representation. **10hours**

Searching: linear search and binary search **5hours**

Sorting: bubble, insertion, selection, quick and merge sort. **5hours**

Books:

1. Data Structure using C- A.M. Tanenbaum (PHI)

BCA 2011: Data Structures C (Practical)

Credit :1 30Hours

All programs should be developed in C

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. (ii) 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.

BCA 2021: Mathematics-II

This course introduces the students to the fundamental concepts and topics of probability and statistics, whose knowledge is important in other computer science courses.

Course Learning Outcomes After successful completion of this course, the student will be able to:

- **Use probability theory to evaluate the probability of real-world events.**
- **Describe discrete and continuous probability distribution functions and generate random numbers from the given distributions.**
- **Find the distance between two probability distributions**
- **Define and quantify the information contained in the data.**
- **Describe various algorithms of Numerical Methods.**

Credit :4 60Hours

Probability and Statistics: Permutation and Combinations, Probability, Classical definition of probability. Conditional probability. Statistical independence of events. Random variable and its expectation and variance, joint dispersion of attributes. 10 hours

Collection and presentation of data. Frequency distribution. Measures of central tendency. Measures of dispersion. Binomial, Poission and Normal distribution. 5 hours

Bivariate Frequency Distributions (scatter Diagram, Correlation coefficient and its properties, regression lines, correlation index and correlation ratio, rank correlation). 5 hours

Multiple linear regression, multiple correlation, partial correlation (for 3 variables only). 5 hours

Random sampling, expectations and standard error of sampling mean. Expectation and standard error of sampling proportions. 5 hours

Test of significance based on t, F, and CHI square distribution. 5 hours

Numerical Methods and Algorithms Solution of non-linear equations: Bisection, Newton-Raphson, Regular-Falsi and Secant method. Interpolation and approximation- Lagrange Interpolation, Newton's Forward Interpolation and Newton's backward Interpolation methods. Integration: Trapizoidal and Simpson's 1/3 rules. Solution of linear equations: Gaussian elimination, Gauss Sheidal method Solution of different equations; Eular's, Taylor's series, Runge-kutta (order-2) 25 hours

Books:

1. C Language and Numerical Methods C Xaviers, New Age International

2. Fundamentals of Statistics – Goon, Gupta, DasGupta

Multi/Interdisciplinary courses
(For Other discipline)
Semester-II

BCA 2031: Introduction to Internet

Credit: 03 **45 Hours**

This course is intended to teach the basics involved in publishing content on the World Wide Web. This includes the 'language of the Web' – HTML and the fundamental principles of how the Internet and the Web function.

Course Learning Outcomes

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

i. Discuss elementary Internet concepts and history.
ii. Make a successful Internet connection.
iii. Demonstrate simple principles of Internet Protocol (IP) addressing.
iv. Use and customize a web browser.
v. Comprehend the basics of the internet and web terminologies.

Introduction : Evolution of Internet, concept of Intranet and Internet, Applications of Internet, Types of Connectivity such as dial – up, leased, VSAT. etc., Internet Server and Clients module in various Operating Systems.(5 Hrs)

Usenet and Internet Relay Chart Introduction to World Wide Web: Evolution of WWW, Basics Features, WWW Browsers, WWW servers, HTTP & URL's. (5 Hrs)

Browsers: Basic features, Bookmarks, history. Progress indicators, Personalization of Browsers, Printing displayed pages and forms, Saving Web pages, Netscape Communicators, Internet Explorer, Search and Downloads.(5 Hrs)

Search Engines: Technology overview, Popular Search Engines.How to register a website in search engine. (5 Hrs)

Internet Security: Overview of Internet Security threats, Firewalls, Introduction to AAA (5 Hrs)

HTML: (20 Hrs)

- **Unit-I: Introduction**

- **Unit-II: The Basics**

- The Head, the Body
- Colors, Attributes
- Lists, ordered and unordered
  - **Unit-III: Links**

- Introduction
- Relative Links,
- Absolute Links
- Link Attributes
- Using the ID Attribute to Link Within a Document

- **Unit-IV: Images**

- Putting an Image on a Page
- Putting an Image in the Background

- **Unit V: – Tables**

- Creating a Table
- Table Headers Captions
- Spanning Multiple Columns
- Styling Table

Reference Books

1. Internetworking with TCP/IP – by D.E.Comer, PHI
2. Introduction to HTML and CSS -- O'Reilly

BCA 2051 : Python Programming

Course Learning Outcomes: After successful completion of the Course a student will be able to:

- **Learn the basic knowledge of Python. • Students will be able to acquire programming skills in core Python.**
- **Students will be able to acquire Object Oriented Skills in Python.**
- **Students will be able to solve problems requiring the writing of well-documented programs in The Python language, including use of the logical constructs of that language.**

Credit :3 90Hours

Planning the Computer Program: Concept of problem solving, Problem definition, Program design, Debugging, Types of errors in programming, Documentation. **06hours**

Techniques of Problem Solving: Flowcharting, decision table, algorithms, Structured programming concepts, Programming methodologies viz. top-down and bottom-up programming**.** **10hours**

Overview of Programming: Structure of a Python Program, Elements of Python **04hours**

Introduction to Python: Python Interpreter, Using Python as calculator, Python shell, Indentation. Atoms, Identifiers and keywords, Literals, Strings, Operators (Arithmetic operator,

Relational operator, Logical or Boolean operator, Assignment, Operator, Ternary operator, Bit wise operator, Increment or Decrement operator). **10hours**

Creating Python Programs: Input and Output Statements, Control statements (Branching, Looping, Conditional Statement, Exit function, Difference between break, continue and pass.), Defining Functions, default arguments, Errors and Exceptions. **10hours**

Iteration and Recursion: Conditional execution, Alternative execution, Nested conditionals, The return statement, Recursion, Stack diagrams for recursive functions, Multiple assignment, The while statement, Tables, Two-dimensional tables Strings and Lists: String as a compound data type, Length, Traversal and the for loop, String slices, String comparison, A find function, Looping and counting, List values, Accessing elements, List length, List membership, Lists and for loops, List operations, List deletion. Cloning lists, Nested lists Object Oriented Programming: Introduction to Classes, Objects and Methods, Standard Libraries. Data Structures: Arrays, list, set, stacks and queues. **20hours**

Searching and Sorting: Linear and Binary Search, Bubble, Selection and Insertion sorting. **04 hours**

Strings and Lists: String as a compound data type, Length, Traversal and the for loop, String slices, String comparison, A find function, Looping and counting, List values, Accessingelements, List length, List membership, Lists and for loops, List operations, List deletion. Cloning lists, Nested lists. **14hours**

Object Oriented Programming: Introduction to Classes, Objects and Methods, Standard Libraries. **04hours**

Data Structures: Arrays, list, set, stacks and queues. **04hours**

Searching and Sorting: Linear and Binary Search, Bubble, Selection and Insertion sorting. **04hours**

Software Lab Based on Python:

Section: A (Simple programs)

 Write a menu driven program to convert the given temperature from Fahrenheit to Celsius and vice versa depending upon users choice.

 WAP to calculate total marks, percentage and grade of a student. Marks obtained in each of the three subjects are to be input by the user. Assign grades according to the following criteria :

Grade A: Percentage >=80

Grade B: Percentage>=70 and <80Grade C:

Percentage>=60 and <70Grade D:

Percentage>=40 and <60 Grade E:

Percentage<40

Section: B (Visual Python):

All the programs should be written using user defined functions, wherever possible.

1. Write a menu-driven program to create mathematical 3D

objects I. curve

II.

sphere

III. cone

IV. arrow

V. ring

VI. cylinder.

2. WAP to read n integers and displaythem as a histogram.

3. WAP to displaysine, cosine, polynomial and exponential curves.

4. WAP 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.

5. WAP to calculate the mass m in a chemical reaction. The mass m (in gms) disintegrates according to the formula m=60/(t+2), where t is the time in hours. Sketch a graph for t vs. m, where t>=0.

6. A population of 1000 bacteria is introduced into a nutrient medium. The population p Grows as follows:

P(t) = (15000(1+t))/(15+ e)

where the time t is measured in hours. WAP to determine the size of the population at given time t and plot a graph for P vs t for the specified time interval.

7. Input initial velocity and acceleration, and plot the following graphs depicting equations of motion:

I. velocitywrt time (v=u+at)

II. distance wrt time (s=u*t+0.5*a*t*t)

III. distance wrt velocity (s=(v*v-u*u)/2*a)

Books:

(i) T. Budd, Exploring Python, TMH, 1st Ed, 2011

ii. How to think like a computer scientist : learning with Python / Allen Downey, Jeffrey Elkner, Chris

Meyers. 1st Edition – Freely available online.2012

1. http://docs.python.org/3/tutorial/index.html

2. http://interactivepython.org/courselib/static/pythonds

SEMESTER – III
Major/DS course (Core) BCA 3011
Object Oriented Programming Using JAVA

Credit : *5*

Course Learning Outcomes: After successful completion of the Course

- **Able to solve real world problems using OOP techniques.**
- **Able to understand the use of abstract classes.**
- **Able to solve problems using java collection framework and I/o classes.**
- **Able to develop multithreaded applications with synchronization.**
- **Able to develop applets for web applications**

Introduction: JAVA as internet language. A first simple program. Entering the program, Compiling the program, control statements, using blocks of code, lexical issues-white space, identifiers, literals, comments, separators, The Java keyword-The java class libraries, data types, variables and arrays, the simple types, integers-byte, short, int, long, floating point types-float, double, characters, Booleans. A closer look at literals-integer literals, floating point literals, Boolean literals, character literals, string literals, variables – declaring a variable, dynamic initialization, the scope and lifetime of variables, type conversion and casting – java's automatics conversions, casting incompatible types, automatic type promotion in expressions, the type promotion rules, arrays-one dimensional arrays, multidimensional arrays, alternative array declaration syntax. Operators-assignment operators, increment and decrement, the bitwise operators, relational operators, Boolean logical operators, the assignment operator, the ? operator, operator precedence, using parentheses, control statements-Java's selection statements, if, switch, interaction statements-while, do-while, for, some for loop variations, nested loops, jump statements-using break, using continue, return. [20L]

Introduction Classes: class fundamentals, the general form of a class, a simple class, declaring objects, a closer look at new, assigning object reference variables, introducing methods, adding a method, the box class, returning a value, adding a method to the box class, constructors, parameterized constructors, the this keyword, instance variable hiding, garbage collection, the finalize method, a stack class, a closer look at methods and classes, overloading methods, overloading constructors. Using object as parameters, argument passing, returning objects, introducing access control, understanding static, introducing final, arrays revised., exploring the string class, using command line arguments, inheritance, inheritance basics, a more practical example, a superclass variable can reference a subclass object, using super, using super to call superclass constructors, a second use for super, creating a multilevel hierarchy, when constructor are called, method overriding, dynamic method dispatch, overridden methods, applying methods overriding, using abstract classes, using final with inheritance, using final to prevent overriding, using final to prevent inheritance, the object class. [20L]

Package and interfaces: Defining a package, understanding class path, a short package example, access protection, an access example, importing packages- interface-defining an

interface, implementing interfaces, applying interfaces, variables in interfaces, interface can be extended.[7L]

Exception Handling: Exception fundamentals, exception types, uncaught exceptions, using try and catch, displaying a description of an exception, multiple catch clauses, nested try statements, throw, throws, finally, java's built-in exceptions-creating you own exception subclasses, using exceptions, multithreaded programming. [8L]

Threads: The java's thread model, thread priorities, synchronization, message, the thread class and the runnable interface, the main thread, creating a thread, implementing runnable, extending thread, closing an approach, creating multiple thread, thread priorities, synchronization, using synchronized methods, the synchronized statement, interthread communication, using multithreading. [10L]

Applets: Concept, and use of applet, I/O basics, streams, applet fundamentals, writing simple applet program. [5L]

Stream classes: The predefined streams, reading console input, writing console output, reading and writing files, the transient and volatile modifiers, using instance of native methods, problem with native methods. [5L]

String handling: The string constructor, string length, special string operations-string literals, string concatenation, string concatenation with other data types, string conversion, character extraction, string comparison, searching strings, modifying a string-data conversion using string buffer constructors. [10L]

Wrappers: Number, double and float, integer and long, character, Boolean, process, runtime, memory management, executing other programs, system-using current time limits to time, program execution, using array copy, environment properties, object, using clone and the cloneable interface, class, class loader, math-transcendental functions, exponential functions, rounding functions, miscellaneous math methods, classes, input stream, output stream file input stream, file output stream. [10L]

Java Database Connectivity (JDBC): Implementation of simple system using JDBC. [5L]

Books: 1. Programming with JAVA – E.Balagurusamy, TMH

Major/DS course (Core) BCA 3012
Operating System

Credit : *5*

Course Learning Outcomes: After successful completion of the Course the student will:

- know basic components of an operating system.
- comprehend how an operating system virtualises CPU and memory
- discuss various scheduling and swapping policies

- get to know how an operating system protects the computer system

Operating system as an extended machine and a resource manager, operating system conceptsprocess, files, shell, Operating system structure: monolithic system, layered systems, virtual machines, client server model. Idea of multiprogramming, multiprocessing, batch processing and time sharing. Real time systems [30L]

Concurrent processes: Critical section problem, Semaphores & Synchronization. [10%]

CPU scheduling: Scheduling concepts and algorithms [15L]

Memory management: Static & dynamic partitioning, Dynamic relocation, Paging & demand paging memory management, Virtual memory, Replacement algorithm, Segmented memory management, Thrashing. [15L]

Device management: Scheduling concept and algorithm, spooling. [10L]

Deadlock detection, prevention and avoidance. [10L]

File management: File concept, access methods, allocation methods, Directory concept. [10L]

Books:
1. Modern Operating Systems- A.S. Tanenbaum (PHI) 2. UNIX – S.Das, TMH

Multi/Interdisciplinary BCA 3031
Introduction to Cyber Security

Credit: 3

Course learning outcome

Upon completion of the degree program, students will be able to:

1. Analyze and evaluate the cyber security needs of an organization.
2. Conduct a cyber security risk assessment.
3. Measure the performance and troubleshoot cyber security systems.
4. Implement cyber security solutions.
5. Be able to use cyber security, information assurance, and cyber/computer forensics software/tools.
6. Identify the key cyber security vendors in the marketplace.
7. Design and develop a security architecture for an organization.

Introduction to Cyber Space

History of Internet , Cyber Crime , Information Security ,Computer Ethics and Security , Choosing the Best Browser according to the requirement and email security, Guidelines to choose web browsers , Securing web browser , Antivirus , Email security

Guidelines for secure password and wi-fi security

Guidelines for setting up a Secure password ,Two-steps authentication ,Password Manager ,Wi-Fi Security

Guidelines for social media and basic Windows security

Guidelines for social media security ,Tips and best practices for safer Social Networking , Basic Security for Windows ,User Account Password

Smartphone security guidelines

Introduction to mobile phones , Smartphone Security ,Android Security ,IOS Security

Cyber Security Initiatives in India

Counter Cyber Security Initiatives in India , Cyber Security Exercise ,Cyber Security Incident Handling

Cyber Security Assurance

Online Banking, Credit Card and UPI Security

Online Banking Security , Mobile Banking Security , Security of Debit and Credit Card , UPI Security

Micro ATM, e-wallet and POS Security

Security of Micro ATMs , e-wallet Security Guidelines ,Security Guidelines for Point of Sales(POS)

Social Engineering

Social Engineering , Types of Social Engineering ,How Cyber Criminal Works ,How to prevent for being a victim of Cyber Crime

Cyber Security Threat Landscape and Techniques

Cyber Security Threat Landscape , Emerging Cyber Security Threats , Cyber Security Techniques ,Firewall

IT Security Act and Misc. Topics

IT Act , Hackers-Attacker-Countermeasures ,Web Application Security ,Digital Infrastructure Security ,Defensive Programming

Information Destroying and Recovery Tools

Recovering from Information Loss , Destroying Sensitive Information , CCleaner for Windows

Reference Books:

i) Cybersecurity: The Beginner's Guide: A comprehensive guide to getting started in cybersecurity, Dr. Erdal Ozkaya

ii) Introduction to Cyber Security: concepts, principles, technologies and practices ,Ajay Singh

Skill Enhancement Course (SEC) BCA 3051
Unix and Shell Programming

Credit : 3

On completion of this module the student should be able to:

- Identify and use UNIX/Linux utilities to create and manage simple file processing operations, organize directory structures with appropriate security, and develop shell scripts to perform more complex tasks.
- Effectively use the UNIX/Linux system to accomplish typical personal, office, technical, and software development tasks.

Introduction, Brief history. Unix Components/Architecture. Features of Unix. The UNIX Environment and UNIX Structure. The login prompt. General features of Unix commands/ command structure. Command arguments and options. Understanding of some basic commands such as echo, printf, ls, who, date, passwd, cal, Combining commands. Meaning of Internal and external commands. The type command: knowing the type of a command and locating it. The man command knowing more about Unix commands and using Unix online manual pages. The man with keyword option and whatis. The more command and using it with other commands. Knowing the user terminal, displaying its characteristics and setting characteristics. The root login. Becoming the super user: su command. The /etc/passwd and /etc/shadow files. Commands to add, modify and delete users. Unix files. Naming files. Basic file types/categories. Organization of files. Hidden files. Standard directories. Parent child relationship. The home directory and the HOME variable. Reaching required files- the PATH variable, manipulating the PATH, Relative and absolute pathnames. Directory commands – pwd, cd, mkdir, rmdir commands. The dot (.) and double dots (..) notations to represent present and parent directories and their usage in relative path names. File related commands – cat, mv, rm, cp, wc and od commands. File attributes and permissions and knowing them. The ls command with options. Changing file permissions: the relative and absolute permissions changing methods. Recursively changing file permissions. Directory permissions. [35L]
The vi editor. Basics. The .exrc file. Different ways of invoking and quitting vi. Different modes of vi. Input mode commands. Command mode commands. The ex mode commands. Illustrative examples Navigation commands. Repeat command. Pattern searching. The search and replace command. The set, map and abbr commands. Simple examples using these commands. The shells interpretive cycle. Wild cards and file name generation. Removing the special meanings of wild cards. Three standard files and redirection. Connecting commands: Pipe. Splitting the output: tee. Command substitution. Basic and Extended regular expressions. The grep, egrep. Typical examples involving different regular expressions. Shell programming. Ordinary and environment variables. The .profile. Read and readonly commands. Command line arguments. exit and exit status of a command. Logical operators for conditional execution. The test command and its shortcut. The if, while, for and case control statements. The set and

shift commands and handling positional parameters. The here (<<) document and trap command. Simple shell program examples. File inodes and the inode structure. [35L]

Software Lab based on Unix and shell Programming [30L]

1. Usage of following commands: ls, pwd, tty, cat, who, who am I, rm, mkdir, rmdir, touch, cd.
2. Usage of following commands: cal, cat(append), cat(concatenate), mv, cp, man, date.
3. Usage of following commands: chmod, grep, tput (clear, highlight), bc.
4. Write a shell script to check if the number entered at the command line is prime or not.
5. Write a shell script to modify "cal" command to display calendars of the specified months.
6. Write a shell script to modify "cal" command to display calendars of the specified range of months.
7. Write a shell script to accept a login name. If not a valid login name display message – "Entered login name is invalid".
8. Write a shell script to display date in the mm/dd/yy format.
9. Write a shell script to display on the screen sorted output of "who" command along with the total number of users.
10. Write a shell script to display the multiplication table any number,
11. Write a shell script to compare two files and if found equal asks the user to delete the duplicate file.
12. Write a shell script to find the sum of digits of a given number.
13. Write a shell script to merge the contents of three files, sort the contents and then display them page by page.
14. Write a shell script to find the LCD(least common divisor) of two numbers.
15. Write a shell script to perform the tasks of basic calculator.
16. Write a shell script to find the power of a given number.
17. Write a shell script to find the factorial of a given number.
18. Write a shell script to check whether the number is Armstrong or not.
19. Write a shell script to check whether the file have all the permissions or not.
20. Program to show the pyramid of special character "*".

Books:

1. 1.Sumitabha Das., Unix Concepts and Applications., 4th Edition., Tata McGraw Hill
2. Behrouz A. Forouzan, Richard F. Gilberg : UNIX and Shell Programming- Cengage Learning – India Edition. 2009.
3. M.G. Venkatesh Murthy: UNIX & Shell Programming, Pearson Education.
4. 4.Richard Blum , Christine Bresnahan : Linux Command Line and Shell Scripting Bible, 2ndEdition , Wiley,2014

SEMESTER – IV

Major/DS course (Core) BCA 4011
Digital Logic

Credit : *5*

Course Learning Outcomes :At the end of this course student will:

- Understand various types of number systems and their conversions.
- Simplify the Boolean expressions and apply the Boolean theorems through logical gates
- Design and implement variety of logical devices using combinational circuits concepts.
- Demonstrate and compare the construction of programmable logic devices and different types of ROM
- Analyze sequential circuits like Registers and Counters using flip-flops

Number systems: Positional number systems; Binary, Octal , Hexadecimal and Decimal number systems; conversion of a number in one system to the other; Representation of signed numberssigned magnitude, one's complement, 2's complement representation techniques, Merits of 2's complement representation scheme; Various binary codes- BCD, excess -3, Gray code; Binary arithmetic- addition, subtraction, multiplication and division of unsigned binary numbers. [5L]

Logic gates: Basic logic operations- logical sum(or), logical product (AND), complementation (not), Anti coincidence (EX-OR)and coincidence (EX-NOR) operations: Truth tables of Basic gates; Boolean Variables and Expressions; Demorgan's theorem; Universal gates- NAND and NOR; Boolean expressions Simplification- Algebraic technique, Karnaugh map technique, 3 variable and 4 variable Karnaugh map. [15L]

Combinational Circuits: Half adder, full adder, binary magnitude comparator, adder /subtractor circuits, multiplexer and demultiplexer circuits, BCD adder/ subtractor; ALU; parity generators, code converters, priority encoders, PLAs. [30L]

Sequential circuits: flip- flops, - RS, clocked RS, D, JK, T flipflops,: Race condition, Master Slave JK: Registers- universal shift registers; Counters- Binary, decade; modulo-r divider; Practical IC's; Sequential Machine design. [30L]

Memory Circuits- Qualitative discussion of memory circuits; Classification- Read only, read write, Sequential access, random access; ROM- PROM, EPROM; Static and Dynamic RAM (qualitative) ; Basic idea on CCD memory and magnetic memory: CD-ROM: Structure and use. [10L]

Associative memory, cache memory organization, Virtual memory organization. [10L]

Books:

1. Digital Principles and Applications: Malvino and Leach
2. Modern Digital Electronics : R.P. Jain

3. Digital Circuits & Design – S.Salivahanan, S.Arivazhagan – Vikas Publishing House Pvt Ltd.
4. Digital logic & Computer Design- M.Mano- Prentice Hall of India.

Major/DS course (Core) BCA 4012
Design and Analysis of Algorithms

Credit : *5*

Course Learning Outcomes: On successful completion of this course, the student will be able to:

- Given an algorithm, identify the problem it solves.
- Write algorithms choosing the best one or a combination of two or more of the algorithm design techniques: Iterative, divide-n-conquer, Dynamic Programming using appropriate data structures.
- Write proofs for correctness of algorithms.

Introduction Basic Design and Analysis techniques of Algorithms, Correctness of Algorithm. (5 L)

Algorithm Design Techniques Iterative techniques, Divide and Conquer (Karatsuba integer multiplication, Strassen‟s matrix multiplication, etc.), Dynamic Programming (Convex hull, DFT & FFT), Greedy Algorithms (Knapsack, Denomination of notes/coins), Scheduling. (8 L)

Sorting and Searching Techniques Elementary sorting techniques– Merge Sort, Advanced Sorting techniques - Heap Sort, Quick Sort, Sorting in Linear Time - Bucket Sort, Radix Sort and Count Sort, Searching Techniques, Medians & Order Statistics, complexity analysis. (17 L)

Lower Bounding Techniques: Decision Trees (2 L)

Balanced Trees: AVL Trees, Red-Black Trees (10 L)

Advanced Analysis Technique: Amortized analysis (5 L)

Graphs Graph Algorithms–Breadth First Search, Depth First Search and its Applications, Minimum Spanning Trees (Prim, Kruskal), Shortest Paths (Dijkstra, Bellman-Ford, Floyd-Warshall). (5 L)

String Processing: String Matching, Edit Distance, Document distance, Rabin-Karp, KMP, Boyer-Moore Techniques (5 L)
Cryptographic algorithms and Overview of complexity Classes: Basic cryptographic algorithms like DSA, RSA; P, NP, NP-Hard, NP-Complete (3 L)

Recommended Books:
1. T.H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein Introduction to Algorithms, PHI, 3rd Edition 2009

2. Sarabasse& A.V. Gelder Computer Algorithm – Introduction to Design and Analysis, Publisher – Pearson 3rd Edition 1999

Major/DS course (Core) BCA 4013
Software Engineering

Credit : *5*

Course Learning Outcomes:

- Students will be able to decompose the given project in various phases of a lifecycle.
- Students will be able to choose appropriate process model depending on the user requirements.
- Students can apply the knowledge, techniques, and skills in the development of a software product.

Software Engineering Fundamentals: Definition of software product, Software Engineering Paradigms; Software engineering, Knowledge engineering, and End user development approaches. [8%]

System Analysis: An abstraction, Partitioning and projection, Systems specification, Software Requirements Specification (SRS) standards, Formal Specification methods, Specification tools, Flow based, Data based and Object – Oriented Analysis. [15%]

System Documentation: Principles of system documentation, types of documentation and their importance [5%]

System Planning: Data and fact gathering techniques-Interviewing, communications, presentations and site visit. Feasibility study, feasibility reports, prototyping, cost-benefit analysis-tools and techniques. [8%]

Systems Design:Idealized and constrained design, Process oriented design (Game and Sarson and Yourdon notations), Data oriented design (Warnier–Orr, E-R modelling), Object oriented design (Booch approach), Cohesion and Coupling, Design matrices, Design documentation standard. [25%]

Role of CASE Tools: Relevance of CASE Tools, High-end and Low-end CASE Tools.[7%]
Coding and Programming : Choice of programming languages, Mixed language programming and cell semantics ,Re-engineering legacy systems, Coding standard. [5%]

Software Quality and testing: Software quality assurance .Types of Software Testing (White Box and Black Box Testing, Unit Testing, Integration Testing, Verification and Validation of Software) , Debugging and Software Reliability analysis , Software quality and matrices, Software maturity model and extensions. [10%]

Software Cost and Time estimation: Functions points , Issues in software cost estimation : Introduction to the Rayleigh curve, Algorithmic cost models (COCOMO, Putnam- Slim, Watson, and Felix), Other approaches to software cost and Size estimation (software complexity, Delphi , costing by analogy). [10%]

Software Project Management : Planning software , projects, Work breakdown structures, Integrating software design and project planning ,Software project teams, Projecting monitoring and control. [7%]

Books :

1. Software Engineering, A practioner's Approach- R. S. Pressman (Mc-Graw Hill Inc)
2. An Integrated Approach to Software Engineering-P.Jalote (Narosa Publication House)

Minor Course BCA 4021
Discrete Structure

Credit : 4

Course Learning Outcomes: students learn:

- To develop understanding of Logic Sets and Functions.
- To use mathematical reasoning techniques including induction and recursion
- To understand and apply counting techniques to the representation and characterization of relational concepts.
- To develop an understanding of how graph and tree concepts are used to solve problems arising in the computer science.

Introduction: Sets - finite and Infinite sets, uncountably 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. (15 L)

Growth of Functions: Asymptotic Notations, Summation formulas and properties, Bounding Summations, approximation by Integrals. (8 L)

Recurrences: Recurrence Relations, Generating functions, Linear Recurrence Relations with constant coefficients and their solution, Substitution Method, Recurrence Trees, Master Theorem. (10 L)

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. (15 L)

Propositional Logic: Logical Connectives, Well-formed Formulas, Tautologies, Equivalences, Inference Theory. (12 L)

Recommended Books:

1. C.L. Liu , D.P. Mahopatra, Elements of Discrete mathematics, 2nd Edition , Tata McGraw Hill, 1985, 2. Kenneth Rosen, Discrete Mathematics and Its Applications, Sixth Edition ,McGraw Hill 2006
3. T.H. Coremen, C.E. Leiserson, R. L. Rivest, Introduction to algorithms, 3rd edition Prentice Hall on India, 2009
4. M. O. Albertson and J. P. Hutchinson, Discrete Mathematics with Algorithms , John wiley Publication, 1988

5. J. L. Hein, Discrete Structures, Logic, and Computability, 3rd Edition, Jones and Bartlett Publishers, 2009
6. D.J. Hunter, Essentials of Discrete Mathematics

Minor Course BCA 4022
Principles of Management

Credit : 4
Course Learning Outcomes: Students learn:

- Fundamental knowledge and exposure to Theories and Concept in the Field of Management.
- To develop the knowledge of business and management principles.
- Decision thinking and problem skills.
- To teach a sense of responsibility and a capacity for business management.
- To Explain the financial concepts used in making business decision.

Meaning and Role of Management Information System- Introduction, Definition, System? Approach. Development of Organizational Theory, Management & Organizational Behaviour, Management, Information & System Approach. [25L]

Data Processing- Operation of Manual Information System, Components of Computer System, Conversion of Manual to Computer Based Systems, Data Bank Concept, Types of Computer Based Applications. Information System for Decision Making- Evolution of Information System, Decision Making & Management Information System. [25L]

Strategic & Project Planning for Management Information System- Business Planning, Management Information System Responses, Management Information System Planning- General & Details. Conceptual System Design- Define Problem, Set System Objective, Establish System Constraints, Determine Information Needs & Sources, Develop Alternative Conceptual Design & Documentation, Prepare the Design Report. [25L]

Detailed System Design- Aim, Project Management, Define Subsystem, Input, Output & Process Design, System Testing, Software & Hardware selection, Documentation of Detailed Design. Pitfalls in Management Information Systems. [25L]

Text & Reference Books:
1. Robert G. Murdick, Joel E. Ross, James R. Claggett, Information System for Modern Management. 2. Surendra Basandra, Computers Today

Semester V

BCA 5011: Database Management System

Major/DS Course (Core)

Credit:04 **60 hours**

Course Objective
The objective of the course is to present an introduction to database management systems, with an emphasis on how to organize, maintain and retrieve - efficiently, and effectively - information from a DBMS.

Course Learning Outcomes
Upon successful completion of the course, the student will be able to

- Understand the basic concepts of database management systems.
- Apply SQL to find solutions to a broad range of queries.
- Apply normalization techniques to improve database design.
- Analyze a given database application scenario to use ER model for conceptual design of the database.

Syllabus

1.Introduction
Characteristics of database approach, data models, database system architecture and data independence.

2.Entity Relationship (ER) Modelling
Entity types, relationships, constraints.

3.Relation data model
Relational model concepts, relational constraints, relational algebra, SQL queries

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

5. Transaction Processing
ACID properties, concurrency control & recovery

6. File Structure and Indexing
Operations on files, File of Unordered and ordered records, overview of File organizations, indexing structures for files (Primary index, secondary index, clustering index), Multilevel indexing using B and B+ trees.

Books:

1. R. Elmasri, S.B. Navathe, Fundamentals of Database Systems 6th Edition, Pearson Education, 2010.

2. R. Ramakrishanan, J. Gehrke, Database Management Systems 3rd Edition, McGraw-Hill, 2002.

3. Silberschatz, H.F. Korth, S. Sudarshan, Database System Concepts 6th Edition, McGrawHill, 2010.

4. R. Elmasri, S.B. Navathe Database Systems Models, Languages, Design and application Programming, 6th Edition, Pearson Education, 2013

BCA 5011: Database Management System (Practical)

Credit:01 **30 hours**

Create and use the following database schema to answer the given queries.

EMPLOYEE Schema

Field	Type	NULL	KEY	DEFAULT
Eno	Char(3)	NO	PRI	NIL
Ename	Varchar(50)	NO		NIL
Job_type	Varchar(50)	NO		NIL
Manager	Char(3)	Yes	FK	NIL
Hire_date	Date	NO		NIL
Dno	Integer	YES	FK	NIL
Commission	Decimal(10,2)	YES		NIL
Salary	Decimal(7,2)	NO		NIL

DEPARTMENT Schema

Field	Type	NULL	KEY	DEFAULT
Dno	Integer	No	PRI	NULL
Dname	Varchar(50)	Yes		NULL
Location	Varchar(50)	Yes		New Delhi

Query List

1. Query to display Employee Name, Job, Hire Date, Employee Number; for each employee with the Employee Number appearing first.
2. Query to display unique Jobs from the Employee Table.
3. Query to display the Employee Name concatenated by a Job separated by a comma.
4. Query to display all the data from the Employee Table. Separate each Column by a comma and name the said column as THE_OUTPUT.
5. Query to display the Employee Name and Salary of all the employees earning more than $2850.
6. Query to display Employee Name and Department Number for the Employee No= 7900.
7. Query to display Employee Name and Salary for all employees whose salary is not in the range of $1500 and $2850.

8. Query to display Employee Name and Department No. of all the employees in Dept 10 and Dept 30 in the alphabetical order by name.
9. Query to display Name and Hire Date of every Employee who was hired in 1981.
10. Query to display Name and Job of all employees who don„t have a current Manager.
11. Query to display the Name, Salary and Commission for all the employees who earn commission.
12. Sort the data in descending order of Salary and Commission.
13. Query to display Name of all the employees where the third letter of their name is _A„.
35
14. Query to display Name of all employees either have two _R„s or have two _A„s in their name and are either in Dept No = 30 or their Manger„s Employee No = 7788.
15. Query to display Name, Salary and Commission for all employees whose Commission Amount is 14 greater than their Salary increased by 5%.
16. Query to display the Current Date.
17. Query to display Name, Hire Date and Salary Review Date which is the 1st Monday after six months of employment.
18. Query to display Name and calculate the number of months between today and the date each employee was hired.
19. Query to display the following for each employee <E-Name> earns < Salary> monthly but wants < 3 * Current Salary >. Label the Column as Dream Salary.
20. Query to display Name with the 1st letter capitalized and all other letter lower case and length of their name of all the employees whose name starts with _J„, „A„ and _M„.
21. Query to display Name, Hire Date and Day of the week on which the employee started.
22. Query to display Name, Department Name and Department No for all the employees.
23. Query to display Unique Listing of all Jobs that are in Department # 30.
24. Query to display Name, Dept Name of all employees who have an _A„ in their name.
25. Query to display Name, Job, Department No. And Department Name for all the employees working at the Dallas location.
26. Query to display Name and Employee no. Along with their Manger„s Name and the Manager's employee no; along with the Employees„ Name who do not have a Manager.
27. Query to display Name, Dept No. And Salary of any employee whose department No. and salary matches both the department no. And the salary of any employee who earns a commission.
28. Query to display Name and Salaries represented by asterisks, where each asterisk (*) signifies $100.
29. Query to display the Highest, Lowest, Sum and Average Salaries of all the employees
30. Query to display the number of employees performing the same Job type functions.
31. Query to display the no. of managers without listing their names.
32. Query to display the Department Name, Location Name, No. of Employees and the average salary for all employees in that department.
33. Query to display Name and Hire Date for all employees in the same dept. as Blake.
34. Query to display the Employee No. And Name for all employees who earn more than the average salary.
35. Query to display Employee Number and Name for all employees who work in a department with any employee whose name contains a _T„.
36. Query to display the names and salaries of all employees who report to King.
37. Query to display the department no, name and job for all employees in the Sales department.

BCA 5012: Theory of Computation
Major/DS Course (Core)

Credit:05 **75 hours**

Course Objective

The objective of the course is to introduce the mathematical foundations of computation including automata theory; the theory of formal languages and grammars; the notions of algorithm, decidability, complexity, and computability. Students will learn that certain problems do not admit efficient algorithms and identify such problems

Course Learning Outcomes

Upon successful completion of the course, the student will be able to

- To give an overview of the theoretical foundations of computer science from the perspective of formal languages.
- To illustrate finite state machines to solve problems in computing.
- To explain the hierarchy of problems arising in the computer sciences.
- To familiarize Regular grammars, context frees grammar.

Syllabus

Concept of Automation: Definition, concept of sequential circuits, state table & state diagram, concept of synchronous, asynchronous machines.

Finite State Machines: Basic definition, mathematical representation, Moore versus Mealy m/c, capability & limitations of FSM, state equivalence & minimization, machine equivalence, incompletely specified machines, merger graph & compatibility graph, information loss less & inverse machines: testing table & testing graph.

Finite Automation: Preliminaries (strings, alphabets & languages, graphs & trees, set & relations), definition, recognition of a language by an automata - idea of grammar, Deterministic Finite Automata (DFA), Non-Deterministic Finite Automata (NFA), Regular Expressions, Equivalence of DFAs, NFAs, and Regular, Expressions, Non-regular languages, Pumping lemma.

Introduction, definition, derivation trees, simplification, Context-Free Grammar (CFG), Parse Trees.

Pushdown Automata: Definition, moves, Instantaneous Descriptions, Deterministic & Non-Deterministic Push Down Automata (PDA), Acceptance by final state & Empty stack, Equivalence of CFGs and PDAs.

Books:

1.Hopcroft JE. and Ullman JD., —Introduction to Automata Theory, Languages & Computation‖, Narosa.

2.Lewis H. R. and Papadimitrou C. H., —Elements of the theory of Computation‖, P.H.I.

3.Kain, —Theory of Automata & Formal Language‖, McGraw Hill.

4.Kohavi ZVI, —Switching & Finite Automata‖, 2nd Edn., Tata McGraw Hill.

BCA 5013: PHP Programming
Major/DS Course (Core)

Credit:04 **60 hours**

Course Objective

The objective of the course is to understand the fundamental concepts of PHP scripting language, the basic structure of a web application, basics of MySQL database the relationship between the client-side and server-side scripts and creating functional websites and web apps in PHP.

Course Learning Outcomes

Upon successful completion of the course, the student will be able to

- Learn the environment of Server-Side Script.
- Compare and contrast between Client-Side Script & Server-Side Script.
- Learn the use of control structures and numerous native data types with their methods.
- Make Database connectivity between Front End and Back End.
- Develop Dynamic Website that can interact with different kinds of Database Languages.

Syllabus

Introduction to PHP: Open source-PHP – history- features-variables- statements operators-conditional statements-if-switch nesting conditions-merging forms with conditional statements-loops-while-do-for – loop iteration with break and continue.

Arrays and Functions: Arrays: Creating an array- modifying array-processing array-grouping form with arrays- using array functions- creating user defined functions- using files- sessions-cookies executing external programs- Creating sample applications using PHP.

Database Management System: Components of Database system-Definition and benefits of database Data Independence-Three level of database architecture-Database Management system Client server architecture-Distributed processing-Domains-Relations-Integrity constraints-Candidate keys-Primary keys-Foreign keys-Functional dependency(Basic definition)-Normal Forms (INF, 2NF, 3NF, BCNF)-ER model – OOAD model.

MySQL: Effectiveness of MySQL -MySQL Tools-Prerequisites for MySQL connection Databases and tables- MySQL data types-Creating and manipulating tables Insertion, updation and deletion of rows in tables -Retrieving data- Sorting and filtering retrieved data - Advanced data filtering-Data manipulation functions Aggregate functions -Grouping data- Sub queries- Joining Tables- Set operators Full text searching.

PHP with MySQL: Working MySQL with PHP-database connectivity- usage of MYSQL commands in PHP, processing result sets of queries- handling errors-debugging and diagnostic functions validating user input through Database layer and Application layer-formatting query output with Character, Numeric, Date and time –sample database applications.

Books:

1. VIKRAM VASWANI, —PHP and MySQL‖, Tata McGraw-Hill

2. BEN FORTA , ‖MySQL Crash course — SAMS

3. C.J. DATE, —An Introduction to Database Systems‖, Addison Wesley, Sixth Edition.
4.Ramesh Elmasri and Shamkant B Navathe, Fundamentals of Database Systems, Pearson Education.

BCA 5013: PHP Programming (Practical)

Credit:01 **30 hours**

1. Create a PHP page using functions for comparing three integers and print the largest number.
2. Write a function to calculate the factorial of a number (non-negative integer). The function accept the number as an argument.
3. WAP to check whether the given number is prime or not.
4. Create a PHP page which accepts string from user. After submission that page displays the reverse of provided string.
5. Write a PHP function that checks if a string is all lower case.
6. Write a PHP script that checks whether a passed string is palindrome or not? (A palindrome
is word, phrase, or sequence that reads the same backward as forward, e.g., madam or nurses run)
7. WAP to sort an array.
8. Write a PHP script that removes the whitespaces from a string.
Sample string : 'The quick " " brown fox'
Expected Output : Thequick""brownfox
9. Write a PHP script that finds out the sum of first n odd numbers.
10. Create a login page having user name and password. On clicking submit, a welcome message should be displayed if the user is already registered (i.e.name is present in the database) otherwise error message should be displayed.
11. Write a PHP script that checks if a string contains another string.
12. Create a simple 'birthday countdown' script, the script will count the number of days between current day and birth day.

13. Create a script to construct the following pattern, using nested for loop.

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

14. Write a simple PHP program to check that emails are valid.
15. WAP to print first n even numbers.
16. $color = array('white', 'green', 'red")
Write a PHP script which will display the colors in the following
way : Output : white, green, red,
• green
• red
• white
17. Using switch case and dropdown list display a —Hello‖ message depending on the language selected in drop down list.
18. Write a PHP program to print Fibonacci series using recursion.
19. Write a PHP script to replace the first 'the' of the following string with 'That'.
Sample : 'the quick brown fox jumps over the lazy dog.'
Expected Result : That quick brown fox jumps over the lazy dog.

Semester VI
BCA 6011: Data Communication and Networking
Major/DS Course (Core)

Credit:03 **45 hours**

Course Objective
The objective of the course is to equip students with a comprehensive understanding of network fundamentals, protocols, technologies, and architectures, enabling them to design, implement, and manage computer networks.

Course Learning Outcomes
Upon successful completion of the course, the student will be able to

- Understand the concept of Signals, OSI & TCP/IP reference models and functionalities of each layer in these models.
- Apply flow control and error control mechanisms using data link layer protocols.
- Apply the concepts of various routing algorithms and IP addresses to fulfil network requirements of an organization.
- Determine the model of reliable/unreliable communication using various Transport Layer Protocols (UDP, TCP).

Syllabus
Introduction: Communication systems, Analogue data, digital data, Communication channels, Synchronous data, Asynchronous data. Twin wire, Coaxial cable, Radio, VHF and microwaves, satellite links, Optical fiber. Concepts of modulation, ASK, FSK, PSK, Quadrature PSK, Differential PSK.

Introduction to computer network: Structure of computer network, point-to-point multidrop circuits, Data flow and physical circuits, network topologies, topologies and design goals. Hierarchical topology, horizontal topology, star topology, ring topology, mess topology. Telephone network, switched and non-switched options, fundamentals of communication theory, channel speed and bit rate. Voice communication and analogue waveforms, bandwidth and frequency spectrum, connecting the analogue and digital world, digital signals, the modem, asynchronous and synchronous transmission.

Local area networks and wide area networks, connection oriented and connectionless networks, classifications of communication protocols, time division multiple access (TDMA), time division multiplexing (TDM), carrier sense system(collision), token passing, peer-to-peer priority systems, priority slot, carrier sense systems (collision free). Token passing (priority) systems. Layered Protocols and the OSI model Goals of layered protocols, network design problems, communication between layers, introduction to standard organizations and the OSI model, standards organizations, Layers of OSI, OSI status

Polling/Selection Protocols: Character and bit protocols, binary synchronous control (BSC), HDLC, HDLC options HDLC frame format, code transparency and synchronization, HDLC transmission process, HDLC subsets, SDLC, protocol conversion.

Local Area Networks: Primary attributes of a LAN, Broadband and baseband and base LANs, IEEE LAN standards, relationship of the 802 standards to the ISI/CCITT model, connection options with LANs, LLC and MAC protocol data units, LAN topologies and protocols, CSMA/CD and IEEE 802.3, token ring (priority), token bus and IEEE 802.4, Metropolitan Area Networks (MANs), ANSI fiber distributed data interface.

Switching and Routing in Networks: Message switching, packet switching, when and when not to use packet switching, packet routing, packet switching support to circuit switching networks.

TCP/IP: TCP/IP Reference Model and internetworking, example of TCP/IP operations, related protocols, Concept of ports and sockets.

Books:

1. Black , U., " Computer Networks- protocols, standards and Interfaces" , P.H.I.

2. Stallings, W ., " Computer Communication Networks",P.H.I.

3. Tannembaum , A.S., " Computer Networks", P.H.I.

BCA 6011: Data Communication and Networking (Practical)

Credit:01 **30 hours**

All programs should be developed in C/ C++ / Java / Python
1. Simulate Cyclic Redundancy Check (CRC) error detection algorithm for noisy channel.
2. Simulate Hamming-code based error detection & correction algorithm for noisy channel.
3. Simulate and implement stop and wait protocol for noisy channel.
4. Simulate and implement go back N sliding window protocol.
5. Simulate and implement selective repeat sliding window protocol.
6. Simulate and implement MST construction (Prim‟s, Kruskal‟s) for Ethernet
7. Simulate and implement the various routing algorithms (RIP, Distance-Vector routing, Dijkstra‟s, Bellman-Ford, Floyd-Warshall, Flooding).

BCA 6012: Artificial Intelligence
Major/DS Course (Core)

Credit:03 **45 hours**

Course Objective

The objective of the course is to equip students with the knowledge and skills to identify AI applications, apply basic AI techniques, and design intelligent systems that learn from experience, often focusing on machine learning and deep learning fundamentals.

Course Learning Outcomes

Upon successful completion of the course, the student will be able to

- Identify and apply suitable Intelligent agents for various AI applications
- Identify knowledge associated and represent it by ontological engineering to plan a strategy to solve given problem
- Apply the suitable algorithms to solve AI problems
- Implement ideas underlying modern logical inference systems

Syllabus

Scope of Artificial Intelligence, games, theorem proving, natural language processing, vision and speech processing, robotics, expert systems, AI techniques in search and knowledge abstraction

Problem solving; state space search, search space control, heuristic search, hill climbing, branch and bound.

Knowledge representation; predicate logic, rule-based system, structured knowledge representation, semantic net.

Handling uncertainty, Fuzzy sets, probabilistic reasoning.

Learning, learning automation, learning by induction, Neural Networks, Genetic Algorithms Emerging technologies and devices.

Books:

1.Artificial Intelligence, Rich & Knight, TMH

2.Introduction to AI & Expert Systems, Patterson, PHI

3.Neural Networks, Fuzzy Logic & Genetic Algorithms, Rajsekharan, PHI
4.Expert Systems, Giaranto, VIKAS

BCA 6012: Artificial Intelligence (Practical)

Credit:01 **30 hours**

All programs should be developed using Prolog or LISP
1. Write a prolog/lisp program to calculate the sum of two numbers.
2. Write a prolog/lisp program to find the maximum of two numbers.
3. Write a prolog/lisp program to calculate the factorial of a given number.
4. Write a prolog/lisp program to calculate the nth Fibonacci number.
5. Write a prolog/lisp program, insert_nth (item, n, into_list, result) that asserts that result is the list into_list with item inserted as the nth element into every list at all levels.
6. Write a Prolog/lisp program to remove the Nth item from a list.
7. Write a Prolog/lisp program, remove-nth (Before, After) that asserts the After list is the Before list with the removal of every nth item from every list at all levels.
8. Write a Prolog/lisp program to implement append for two lists.
9. Write a Prolog/lisp program to implement palindrome (List).
10. Write a Prolog/lisp program to implement max (X,Y, Max) so that Max is the greater of two
numbers X and Y.
11. Write a Prolog/lisp program to implement maxlist (List, Max) so that Max is the greatest number in the list of numbers List.
12. Write a Prolog/lisp program to implement sumlist (List, Sum) so that Sum is the sum of a given
list of numbers List.
13. Write a Prolog/lisp program to implement two predicates even length (List) and
Odd length (List) so that they are true if their argument is a list of even or odd length respectively.
14. Write a Prolog/lisp program to implement reverse (List, Reversed List) that reverses lists.
15. Write a Prolog/lisp program to implement maxlist (List, Max) so that Max is the greatest number in the list of numbers List using cut predicate.
16. Write a Prolog/lisp program to implement GCD of two numbers.
17. Write a prolog/lisp program that implements Semantic Networks/Frame Structures.

BCA 6013: Microprocessor
Major/DS Course (Core)

Credit:03 **45 hours**

Course Objective

The course objective is to introduce the basic concepts of microprocessor and to develop in students the assembly language programming skills and real time applications of Microprocessor as well as microcontroller.

Course Learning Outcomes

Upon successful completion of the course, the student will be able to

- Understand the basic architecture of 8085 and 8086.
- Impart the knowledge about the instruction set.
- Understand the basic idea about the data transfer schemes and its applications
- Develop skill in simple program writing for INTEL 8085 and INTEL 8086.

Syllabus

Microprocessors: 8-bit microprocessor architecture; 8085 pin description

Programmers model of 8085, addressing modes of 8085; Instruction set of 8085; Assembly language program for 8085

Memory interfacing; I/O interfacing; Peripheral ICs; I/O memory Interfacing Chips, Bus structure of microprocessor-based systems, bus arbitration, Interrupt handling and DMA operation. Basic idea about microprogramming.

Advanced Microprocessors: Functional description of 8086 microprocessor, software model of 8086/8088; Data addressing modes of 8086; 80x86 family of microprocessor

Comparison of different microprocessors; microprocessors of other families; Assembly language program of 8086 microprocessor.

Books:

1. Microprocessors: A.P.Mathur

2. Microcomputer systems -8086/8088 family: Liu and Gibson

3. Programming the 80286,80386 Computers: B.B. Brey 4. Microprocessors and Interfacing: Hall

BCA 6013: Microprocessor (Practical)

Credit:01 **30 hours**

All programs should be developed for the Intel 8085 Microprocessor.
ASSEMBLY LANGUAGE PROGRAMMING

1. Write a program for 32-bit binary division and multiplication
2. Write a program for 32-bit BCD addition and subtraction
3. Write a program for Linear search and binary search.
4. Write a program to add and subtract two arrays
5. Write a program for binary to BCD conversion
6. Write a program for BCD to binary conversion

BCA 6014: Internet & E-Commerce
Major/DS Course (Core)

Credit:03 **45 hours**

Course Objective

The course objective is to learn the basic concepts of Internet and e-commerce technologies, and gain hands-on experience in setting up systems for Internet and e-commerce applications

Course Learning Outcomes

Upon successful completion of the course, the student will be able to

- The principles of Internet technologies and applications.
- Understand different file transfer protocols and search engines.
- Different payment methodology and security aspects.
- Develop a sense to use the Internet and e-commerce for enhancing daily life and work

Syllabus

Introduction to Internet: Evolution of Internet, concept of Intranet and Internet, Applications of Internet, Types of Connectivity such as dial – up, leased, VSAT. etc., Internet Server and Clients module in various Operating Systems, TCP/IP, Introduction to RFC, Addressing in Internet – IP and Domains, major features of IP, IP datagram, major IP services, IP source routing, value of the transport layer, TCP, major features of TCP, passive and active operation, Internet Service Providers.

E-mail and List-servers: E-mail Networks, E-mail protocols (X.400, SMTP, UUCP), Format of an E- mail message, Description of E-mail Headers, E- mail contents and encoding, E-mail routing, List servers, E-mail clients, POP-3, IMAP-4.

File Transfer Protocol: Introduction to FTP, public domain Software, Types of FTP Servers, FTP clients, Common Commands.

Telnet: Telnet protocol, Server daemon, Telnet clients, Terminal emulation

Usenet and Internet Relay Chart Introduction to World Wide Web: Evolution of WWW, Basics Features, WWW Browsers, WWW servers, HTTP & URL's

WWW Browsers: Basic features, Bookmarks, history. Progress indicators, Personalization of Browsers, Printing displayed pages and forms, Saving Web pages, Netscape Communicators, Internet Explorer, Search and Downloads.

Web Publishing: Technology Overview, Web site planning, where to host your Web site, Multiple sites on one server, Maintaining a Web site, Publishing tools.

HTML: Document overview, Header elements, Section Headings, Block- oriented elements, Lists, Inline elements, Visual Mark-up, Hypertext links, Uniform Resource Locators, (URL's), Images, Forms, Tables, Special characters

Search Engines: Technology overview, Popular Search Engines, how to register a Web site on search engines.

Internet Security: Overview of Internet Security threats, Firewalls, Introduction to AAA.

E-commerce: Introduction to E-commerce, Payment Methodology, Security aspects, Standard in electronic payment. E-commerce and Banking, E-commerce and Retailing.

Books:

1. Internetworking with TCP/IP – by D.E.Comer, PHI

2. E-Commerce-Paul A.Murphy, TMH

Semester VII

BCA 7011: Internet Technology (Theory)

Course Learning Outcomes:

After undergoing the course, Students will be able to:

- Implement modular and efficient programs using Object-Oriented Programming (OOP) principles.
- Develop interactive web interfaces by applying JavaScript data types, control structures, and Event Handling techniques to respond to user actions.
- Establish and manage secure communication between Java applications and relational databases using the JDBC API.
- Design and deploy dynamic, data-driven web pages using Java Server Pages (JSP)**.**
- Utilise Implicit JSP Objects, conditional processing, and error-handling mechanisms to manage session data and share information across requests and users.

Credit: 04 **60 Hours**

Java: Use of Objects, Array and Array List class **(5 HRS)**

JavaScript: Data types, operators, functions, control structures, events and event handling. **(15 HRS)**

JDBC:JDBC Fundamentals, Establishing Connectivity and working with connection interface, working with statements, Creating and Executing SQL Statements, Working with Result Set Objects. **(10 HRS)**

JSP: Introduction to Java Server Pages, HTTP and Servlet Basics, The Problem with Servlets, The Anatomy of a JSP Page, JSP Processing, JSP Application Design with MVC, Setting Up the JSP Environment, Implicit JSP Objects, Conditional Processing, Displaying Values, using an expression to Set an Attribute, Declaring Variables and Methods, Error Handling and Debugging, Sharing Data Between JSP Pages, Requests, and Users, Database Access. **(20 HRS)**

Java Beans: Java Beans Fundamentals, JAR files, Introspection, developing a simple Bean, Connecting toDB **(10 HRS)**

Recommended Books:

1. Ivan Bayross, Web Enabled Commercial Application Development Using Html, Dhtml,javascript, Perl Cgi , BPB Publications, 2009.
2. Cay Horstmann, BIG Java, Wiley Publication, 3rd Edition., 2009
3. Herbert Schildt, Java 7, The Complete Reference, , 8th Edition, 2009.

4. Jim Keogh, The Complete Reference J2EE, TMH, , 2002.
5. O'Reilly, Java Server Pages, Hans Bergsten, Third Edition, 2003.

BCA 7011: Internet Technology (Practical)

Credit: 02 **60 Hours**

Create event driven program for following:
1. Print a table of numbers from 5 to 15 and their squares and cubes using alert.
2. Print the largest of three numbers.
3. Find the factorial of a number n.
4. Enter a list of positive numbers terminated by Zero. Find the sum and average of these numbers.
5. A person deposits Rs 1000 in a fixed account yielding 5% interest. Compute the amount in the account at the end of each year for n years.
6. Read n numbers. Count the number of negative numbers, positive numbers and zeros in the list

BCA7012 : Compiler Design (Theory)

Course Learning Outcomes:

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

- Describe the concepts and different phases of compilation.
- Represent language tokens using regular expressions and context free grammars.
- Describe the working of lexical analyzers.
- Understand the working of different types of parsers and parse a particular string.
- Describe intermediate code representations using syntax trees and DAG's as well as use this knowledge to generate intermediate code in the form of three address code representations.
- Apply optimization techniques to intermediate code and generate machine code for high level language program.
- Use Lex and Yacc automated compiler generation tools.

Credit:5 **75 Hour**

Unit 1 Introduction: Overview of compilation, Phases of a Compiler. (5 Hrs)

Unit 2 **Lexical Analysis:** Role of a Lexical analyzer, Specification and recognition of tokens, Symbol table, Error reporting, Regular expressions and definitions, Lexical Analyzer Generator-Lex.(15 Hrs)

Unit 3 **Syntax Analysis:** CFGs, left recursion, left factoring, Top-down parsing- LL parser, Bottom-up parsing- LR parser, Parser Generator-yacc.(15 Hrs)

Unit 4 Intermediate representations: Syntax Directed Definitions, Evaluation Orders for Syntax Directed Definitions, Intermediate Languages: Syntax Tree,

Three Address Code, Types and Declarations, Translation of Expressions, loops and conditional statements, Type Checking. (15 Hrs)

Unit 5 Storage organization & Code generation: Activation records, stack allocation, Issues in Code Generation — Design of a simple Code Generator. (15 Hrs)

Unit 6 **Code optimization:** Principal sources of optimization, Peephole optimization. (10 Hrs)

Reference Books

1. Aho, A., Lam, M., Sethi, R., & Ullman, J. D. Compilers: Principles, Techniques, and Tools, 2ND edition, Addison Wesley, 2006.
2. V Raghvan, Principles of Compiler Design, TMH, 2010.
3. Santanu Chattopadhayay, Compiler Design, PHI, 2005.

BCA7012 : Compiler Design (Practical)

Credit:2 **60 Hour**

Suggested Practical List

1. Write a Lex program to count the number of lines and characters in the input file.
2. Write a Lex program to count the number of vowels and consonants in a given string
3. Write a Lex program that implements the Caesar cipher: it replaces every letter with the one three letters after in alphabetical order, wrapping around at Z. e.g. a is replaced by d, b by e, and so on z by c.
4. Write a Lex program that finds the longest word (defined as a contiguous string of upper and lower case letters) in the input.
5. Write a Lex program that distinguishes keywords, integers, floats, identifiers, operators, and comments in any simple programming language.
6. Write a Lex program to count the number of words, characters, blank spaces and lines in a C file.
7. Write a Lex specification program that generates a C program which takes a string "abcd" and prints the following output

   abcd

   abc
8. Write a Lex program to recognize a valid arithmetic expression.

BCA 7013 : Advanced Data Structure & Algorithm(Theory)

Course Learning Outcomes:

After undergoing the course, Students will be able to understand:

- Students should develop a sound theoretical understanding of advanced algorithms and practical problem-solving skills using them.
- Students should develop basic knowledge of a wide range of advanced algorithm design techniques including dynamic programming, linear programming, approximation algorithms, and randomized algorithms.
- Students should develop basic advanced algorithm analysis skills for analysing the approximation ratio of approximation algorithms and the probability of randomized algorithms.
- Students should gain a good understanding on a wide range of advanced algorithmic problems, their relations and variants, and application to real-world problems

Credit : 04 **60 Hours**

UNIT-I: Introduction to Advanced Data Structures: Introduction to Data Structures, Linear Data Structures, Binary Trees, insertion and deletion in BST. Query in BST, Thread Binary Trees, Balancing Trees, AVL Trees, Red black Trees, B Tree, Splay Trees, Heap, different data structures for heap, double ended heap, Priority Queue, Data Structure for Disjoint-Sets. K-D Tree, removal of recursion. Introduction to Object Oriented Data Structures. [12 Hrs]

UNIT-II: Analysis of Algorithm: Solving problem using computer, general problem solving approach – STAIR, reduction to known problem, meaning of algorithm, steps in development of algorithm, evolution of algorithms, design by analysis, design by synthesis, algorithm design patterns and frameworks, Correctness of an algorithm; concept of space and time complexity; asymptotic analysis, amortization, randomization and experimental analysis; best, worst and average case, order of growth, asymptotic notations, upper and lower bounds; analysing rate of growth of a function with that of known functions, comparing algorithms w. r. t. order of growth; applying algorithm analysis concepts for complexity analysis of different sorting techniques. [6 Hrs]

UNIT-III: Divide and Conquer: Introduction to top – down approach, divide – and –

conquer approach, identifying problems where divide – and – conquer suits / is unsuitable, analysis of divide and conquer algorithms, recurrences, the master method for solving recurrences. [6Hrs]

UNIT-IV: Dynamic Programming : Principle of optimality, bottom – up approach to problem solving, comparison with top – down approach, application of dynamic programming to solve problems using optimal sequence of decisions, comparison with brute – force method for finding optimal solution, possibility of multiple sequences in dynamic programming, overlapping sub – problems and optimal substructures, avoiding further exploration of sub – optimal decisions using memorizations, examples of problems; correctness of dynamic programming; analysis of dynamic programming. [12 Hrs]

UNIT-V: Greedy Algorithms: Another approach to find optimal solution by generating only one decision sequence (locally best choice when all available choices are exhausted), optimal substructure and greedy choice; greedy algorithms vs. dynamic programming; examples of problems; analysis of greedy algorithm; matroids; correctness of the greedy algorithm on matroids. [6 Hrs]

UNIT-VI: Sorting and Searching Techniques, Elementary sorting techniques– Merge Sort, Advanced Sorting techniques - Heap Sort, Quick Sort, Sorting in Linear Time - Bucket Sort, Radix Sort and Count Sort, Searching Techniques, Medians & Order Statistics, complexity analysis, lower bound on sorting. [6 Hrs]

UNIT-VII: Graph algorithms: Introduction to graphs and their representations, Graph traversal, DFS, BFS and applications, Connectivity, strong connectivity, bi-connectivity, Minimum spanning tree, Shortest paths, all pair shortest path, Matching, Network flow. [6Hrs]

UNIT-VIII: String Algorithms: Naive string matching, Rabin-Karp, Matching with FA, KMP algorithm, Boyer-Moore algorithm, LCS, String Compression Algorithm, Sequence alignment, edit distance, Levenstein distance, Needleman–Wunsch

algorithm, gap, gap penalty. [6Hrs]

Reference Books:

1.Introduction to Algorithms: Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein, MIT

2.Introduction to Algorithms – A Creative Approach: Udi Manber, Addison – Wesley Professional

3.Algorithms: Richard Johnsonbaugh, Marcus Schaefer, Pearson

4.Fundamentals of Computer Algorithms: Ellis Horowitz, Sartaj Sahni, Sanguthevar Rajasekaran, University Press

5.Design and Analysis of Algorithms: Parag Himanshu Dave, Himanshu Bhalchandra Dave, Pearson

6.Analysis of Algorithms: Computational Methods & Mathematical Tools: Micha Hofri, Oxford University Press, USA

BCA 7013: Advanced Data Structure & Algorithms (Practical)

Credit: 02 **60 Hours**

All programs should be developed in C/Java/Python

1. Implement Quick Sort (The program should report the number of comparisons)
2. Implement Merge Sort(The program should report the number of comparisons)
2. Implement Heap Sort(The program should report the number of comparisons)
4. Implement Radix Sort (The program should report the number of comparisons)
5. Implement Counting sort (The program should report the number of comparisons)
6. Implement Threaded Binary Tree, AVL tree , Red Black Tree, B tree, Splay Trees, Heap and perform different operation on them.
7. Implement Naive string matching, Rabin-Karp, Matching with FA, KMP algorithm, Boyer-Moore algorithm, LCS, String Compression Algorithm, Sequence alignment, edit distance, Levenstein distance, Needleman–Wunsch algorithm.
8. Implement Breadth-First Search in a graph.
9. Implement Depth-First Search in a graph.
10. Write a program to determine the minimum spanning tree and shortest path of a graph

BCA 7014 : Soft Computing(Theory)

Course Learning Outcomes:

After undergoing the course, Students will be able to understand:

- Differentiate between Soft Computing and Hard Computing, identifying when to apply probabilistic and heuristic methods over exact algorithms.
- Design Fuzzy Inference Systems (FIS) by selecting appropriate membership functions, defuzzification methods, and rule bases (Mamdani vs. Sugeno).
- Construct Artificial Neural Networks (ANN) for pattern recognition and function approximation using various architectures like Back-propagation and SOM.
- Optimize complex, multi-variable problems using Genetic Algorithms (GA) by simulating biological evolution (crossover, mutation, and selection).
- Integrate these technologies to create "Hybrid Systems" (like Neuro-Fuzzy systems) for real-world automation and decision-making

Credit : 04 **60 Hours**

Module-I (10 Hrs)

Introduction: Introduction to soft computing; introduction to fuzzy sets and fuzzy logic systems; introduction to biological and artificial neural network; introduction to Genetic Algorithm.

Module-II: Fuzzy Logic (20 Hrs)

Classical Sets and Fuzzy Sets and Fuzzy relations : Operations on Classical sets, properties of classical sets, Fuzzy set operations, properties of fuzzy sets, cardinality, operations, and properties of fuzzy relations.

Membership functions : Features of membership functions, standard forms and boundaries, different fuzzification methods.

Fuzzy to Crisp conversions: Lambda Cuts for fuzzy sets, fuzzy Relations, Defuzzification methods.

Classical Logic and Fuzzy Logic: Classical predicate logic, Fuzzy Logic, Approximate reasoning and Fuzzy Implication

Fuzzy Rule based Systems: Linguistic Hedges, Fuzzy Rule based system – Aggregation of fuzzy Rules, Fuzzy Inference System, Mamdani Fuzzy Models – Takagi-Sugeno Fuzzy Models.

Module-III: Neural Networks (20 Hrs)

Introduction to Neural Networks: Advent of Modern Neuroscience, Classical AI and Neural Networks, Biological Neurons and Artificial neural network; model of artificial neuron.

Learning Methods : Hebbian, competitive, Boltzman.

Neural Network models: Perceptron, Adaline and Madaline networks; single layer network; Delta rule and back-propagation and multi layer networks.

Competitive learning networks: Kohonen self organizing networks (SOM), Hebbian learning; Hopfield Networks, Bidirectional Associative Memory (BAM).

Module-IV: Genetic Algorithms (10 Hrs)

Genetic Algorithms: Simple GA, crossover and mutation, Multi-objective Genetic Algorithm (MOGA).

Reference Books:

1. Fuzzy logic with engineering applications, Timothy J. Ross, John Wiley and Sons.
2. S. Rajasekaran and G.A.V.Pai, "Neural Networks, Fuzzy Logic and Genetic Algorithms", PHI
3. Principles of Soft Computing , S N Sivanandam, S. Sumathi, John Wiley & Sons
4. Genetic Algorithms in search, Optimization & Machine Learning by David E. Goldberg
5. Neuro-Fuzzy and Soft computing, Jang, Sun, Mizutani, PHI
6. Neural Networks: A Classroom Approach,1/e by Kumar Satish, TMH,
7. Genetic Algorithms in search, Optimization & Machine Learning by David E. Goldberg, Pearson/PHI
8. A beginners approach to Soft Computing, Samir Roy & Udit Chakraborty,
9. Soft Computing: Fundamentals and Applications, by D. K. Pratihar.

BCA 7014 : Soft Computing(Practical)

Credit : 02 **60 Hours**

All programs should be implemented using C/ Java / Python

Practical exercises based on concepts listed in theory, such as:

1. Applying an ANN to train a model.

2 Apply a Kohonen SOM.

3. Apply a Hopfield network to memorize and rectify patterns.

4. Apply BAM to memorize.

5. Apply Genetic Algorithms to solve an optimization problem.

6. Applying ANN for pattern recognition tasks.

BCA 7021: Management Process and Organizational Behaviour (Theory)

Course Learning Outcomes:

After undergoing the course, Students will be able:

- To explain the importance & role of management in the business organizations.
- To analyze knowledge on the importance of planning and organizing.
- To identify various leadership styles and their suitability to the situation.
- To apply organizational behaviour theories and concepts to individual work experiences.
- To know how to work more effectively in a team environment.

Credit : 03 **45 Hours**

Unit-I: Nature, Scope and process of Management: Concept of Management, Role and importance of management, Functions and levels of Management, Management – a science and an art, Distinction between Management and Administration; Classification of Managerial functions, Managerial and operative functions. **8LH**

Unit-II: Evolution of Management Thought: Early contributors to Management Thought; Emergence of Management Thought; Scientific management; Administrative Theory of Management; Bureaucratic Organization; Behavioural approach (Neo Classical Theory): Human Relations Movement; Behavioral Science approach; Modern approach to management – Systems approach and contingency approach. **10 LH**

Unit-III: Planning and Organizing: Features of planning, Importance of planning, steps in planning, types of planning: Standing plans and single use plans, objectives, policies, strategies, procedures; Decision making; Formal and Informal organizations, organization structure: Line and Staff organization, Difference between line and staff organization, committee organization, project organization, matrix organization (overview), Delegation of Authority, Centralization and Decentralization, merits and demerits of centralization, Departmentalization: Concept and Types; Span of Management. **10LH**

Unit-IV: Staffing and Directions: Features of staffing, elements of staffing; Concept of Direction, Principles of Direction, Supervision, Leadership Functions and Importance, Formal and Informal Leadership, Qualities of a good leader, Leadership

Styles. **6LH**

Unit-V: Coordination and Control: Concept of Coordination; Features of Coordination, Control-Nature of Control, Relationship between Planning and Control, Elements of Control System. **6 LH**

Unit-VI: Introduction to Organizational Behavior: Concept and characteristics of Organizational Behavior (OB), Issues in developing an OB Model; Characteristics of human behavior. **7 LH**

Unit-VII: Personality, perception, learning, value and attitude; concept, meaning, features, and importance; Motivation-Concept, Selected Theories (Maslow, Herzberg and McGregor); Job SatisfactionConcept and Characteristics. **6 LH**

Unit-VIII: Group Dynamics: concept of group, Stages of group development, Types of groups, Work Teams Vs. Work Groups, Group Synergy. **7 LH**

References Book :

1. Management: A Book of Reading – Koontz, Harold and O' Donell, Cyril (Ed.); McGraw Hill Inc., New York.
2. Management – Stoner, James A.F. and Freeman, R. Edward; Prentics Hall of India, New Delhi.
3. Principles and Practics of Management – Chhabra, T.N.; Dhanpat Raj & Co., New Delhi
4. Management Theory and Practice – Gupta, C.B; Sultan Chand and Sons, new Delhi.
5. Management Principles and Practices – Murgan, M.S.; New Age International Publishers, New Delhi
6. Organizational behavior – Robins Stephen P; PHI.
7. Organizational behavior- Fred Luthans; McGraw Hill Inc.
8. Management of Organizational behavior – Harsey, Paul & Kennith H. Blancher; PHI

BCA 7021: Management Process and Organizational Behaviour (Tutorial)

Credit:1 **15 Hour**

Tutorial based on concepts listed in theory.

Semester VIII

BCA 8011:Network Security & Cryptography (Theory)

Course Learning Outcomes:

After undergoing the course, Students will be able :

- To understand basics of Cryptography and Network Security.
- To secure a message over insecure channel by various means.
- To learn about how to maintain the Confidentiality, Integrity and Availability of a data.
- To understand various protocols for network security to protect against the threats in the networks.

Credit: 04 **60 Hours**

Unit I: Introduction to security, brief history of cryptography, understanding attacks, services, mechanisms, security attacks, security services, model for network security, internet standards, Basic principles of good cryptosystems, Modular arithmetic and GF (2), substitution and transposition ciphers, Stream ciphers, Pseudo Random Number generators. [6Hrs]

Unit II: Symmetric block ciphers, DES, Gallois field, polynomial arithmetic, AES, Principles of S Box design, Block cipher design principles, Other Block ciphers. [6Hrs]

Unit III: Introduction to public key cryptography, Number theoretic foundations, public key cryptography principles, RSA encryption system, primality testing, Number theoretic algorithms, Attacks on RSA. [12 Hrs]

Unit IV: Discrete Logarithm and Diffie-Hellman Key exchange, ElGamal system, Digital Signature, RSA and ElGamal based Digital signature, DSA, different attacks on digital signatures. [12 Hrs]

Unit V: Secure hash functions, Understanding Collisions, Secure Hash Algorithms, SHA, HMAC, Key establishment and key management, Digital Certificates. [6Hrs]

Unit VI: Advanced topics on cryptography: Integer factorization, Strong primes, distribution of Primes, Lattice based cryptography, Introduction to Elliptic curves, Introduction to Elliptic curve-based cryptography, Secret Sharing, Zero Knowledge Proof. [10Hrs]

Unit VII: Authentication, e mail security, Kerberos, x.509 directory authentication services, PGP, S/MIME. Overview of IP security, IP security architecture,

authentication header, encapsulating security pay load, combing security associations. [4 Hrs]

Unit VIII: Introduction to web security, web security requirements, SSL & transport layer security, SET network management security. Basics of system security, intruders, viruses-related threats, firewalls, design principles, trusted systems. [4Hrs]

BCA 8011: Network Security & Cryptography (Practical)

Credit: 02 **60 Hours**

1. Implement RSA Algorithm.
2. Implement Elliptic curve-based cryptography
3. Implement SSL/TLS for secure browser communication using OpenSSL
4. Implement Digital Signature Algorithm (DSA) .
5. Implement SSL/TLS for secure browser communication using OpenSSL
6. Setting up Firewall rules using iptables in Linux
7. Implement Secure Hash Algorithm implementation
8. Implement Primality testing using Farmet's theorem.
9. Implement Miller-Rabin Test.
10. Find inverse of an element in a ring using EEA.
11. Implement Modular Arithmetic and Polynomial Arithmetic.

BCA 8012:Computer Graphics (Theory)

Course Learning Outcomes:

After undergoing the course, Students will be able to:

- Define the basic elements of computer graphics and identify its diverse applications across various industries.
- Compare and contrast the architectures of Raster and Random scan display devices and understand the integration of input/output hardware.
- Implement and analyze algorithms for scan conversion of lines, circles, ellipses, and polygon filling, including the handling of thick primitives.
- Apply mathematical principles to perform 2D and 3D transformations and execute line and polygon clipping techniques.

- Construct Viewing Transformations, including Parallel and Perspective projections, while accounting for vanishing points to create 3D depth.

Credit: 03 **45 Hours**

1. Introduction (5 Hrs)
Basic elements of Computer graphics, Applications of Computer Graphics.

2. Graphics Hardware (8 Hrs)
Architecture of Raster and Random scan display devices, input/output devices.

3. Fundamental Techniques in Graphics (10 Hrs)
Raster scan line, circle and ellipse drawing, thick primitives, Polygon filling, line and polygon clipping algorithms, 2D and 3D Geometric Transformations, 2D and 3D Viewing Transformations (Projections- Parallel and Perspective), Vanishing points.

4. Geometric Modeling (8 Hrs)
Representing curves & Surfaces.

5. Visible Surface determination (8 Hrs)
Hidden surface elimination.

6. Surface rendering (6 Hrs)
Illumination and shading models. Basic color models and Computer Animation

Books Recommended:

1. J.D.Foley, A.Van Dan, Feiner, Hughes Computer Graphics Principles & Practice 2nd edition Publication Addison Wesley 1990.
2. D.Hearn, Baker: Computer Graphics, Prentice Hall of India 2008.
3. D.F.Rogers Procedural Elements for Computer Graphics, McGraw Hill 1997.
4. D.F.Rogers, Adams Mathematical Elements for Computer Graphics, McGraw Hill 2nd edition 1989.

BCA 8012:Computer Graphics (Practical)

Credit: 01 **30 Hours**

All programs should be developed using C / Java / Python

1. Write a program to implement DDA, and Bresenham's line drawing algorithm.
2. Write a program to implement Bresenham's , and 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 apply various 2D transformations on a 2D object (use homogenous coordinates).
6. Write a program to apply various 3D transformations on a 3D object and then apply parallel and perspective projection on it.
7. Write a program to draw Hermite/Bezier curve.

BCA 8013: Machine Learning (Theory)

Course Learning Outcomes:

- Define and explain machine learning concepts, types, and basic metrics.
- Implement and apply supervised learning techniques (e.g., KNN, Linear Regression,Logistic Regression).
- Apply unsupervised learning methods (e.g., K-Means, Hierarchical Clustering, Association Rules).
- Develop and evaluate simple machine learning models (e.g., Perceptron, single-layerneural networks).
- Analyze and apply appropriate machine learning algorithms depending on theproblems with some real-world data.
- Implement and evaluate supervised learning techniques, including K-NearestNeighbours, linear regression, and logistic regression, and measure model performanceusing accuracy, precision, recall, and F1 score.

Credit: 03 **45 Hours**

UNIT I: Introduction to Machine Learning (20 Hrs)

Introduction: Definition, History and Application of Machine Learning,
Types of Machine Learning: Supervised, Unsupervised, Semi-Supervised, and Reinforcement Learning.
Labelled and Unlabelled Dataset.
Supervised Learning Tasks: Regression vs. Classification,
Learning Framework: Training, Validation and Testing of ML models. *Performance Evaluation Parameters*: Confusion matrix, Accuracy, Precision, Recall, F1 Score, and AUC.
Deep Neural Network (DNNs): Concept of Deep Learning, Deep Neural Networks (DNNs), Difficulty of training DNNs.

UNIT II: Supervised Learning and Unsupervised Learning (25 Hrs)
Regression: Linear and non-linear Regression, Logistic Regression.
Classification: Naïve Bayes, K-Nearest Neighbors, Decision Trees.

Models: Introduction to Artificial Neural Networks, Perceptron Learning Algorithm, Single Layer Perceptron, Introduction to Support
Vector Machine for linearly separable data.
Clustering: K-Means, Hierarchical Clustering,
DBSCAN, Clustering Validation Measures.
ML Applications: Ethical Considerations in Machine Learning, Case study and Real-world Applications.

Reference Books:

1. Rajiv Chopra (2024), Machine Learning and Machine Intelligence, Khanna PublishingHouse.
2. Jeeva Jose (2023), Introduction to Machine Learning, Khanna Publishing House.
3. Mitchell T. (1997). Machine Learning, First Edition, McGraw-Hill.
4. Kalita, J. K., Bhattacharyya, D. K., & Roy, S. (2023). Fundamentals of Data Science: Theory and Practice. Elsevier. ISBN9780323917780
5. Flach, P. A. (2012). Machine Learning: The Art and Science of Algorithms that Make Sense of Data. Cambridge University Press. ISBN: 9781107422223, 2012.
6.. Duda, R. O., Hart, P. E., Stork, D (2007). Pattern classification (2Ed), John Wiley &Sons, ISBN-13: 978-8126511167.
7. Haykin S. (2009). Neural Networks and Learning Machines, Third Edition, PHI Learning.
8. Chollet, F. (2018). Deep Learning with Python. Manning Publications.

BCA 8013: Machine Learning (Practical)

Credit: 01 **30 Hours**

LAB Experiments(Program should be developed using Python)
The lab experiments may be implemented in Python using relevant ML libraries, and datasets from Kaggle, public repositories, or generated datasets.

Suggested list of Experiments:
1. Implement linear regression on a dataset and visualize the regression line.
2. Implement logistic regression on a binary classification dataset and plot the decision boundary.
3. Implement and evaluate the performance of Decision tree ID3/Cart classifier for any given dataset.
4. Implement and evaluate the performance of the Naive Bayes Classifier on a given dataset.
5. Build and evaluate a random forest classifier using a numerical dataset.
6. Implement a support vector machine for linearly separable classes and visualize the margins and decision boundary.

7. Implement K-Means clustering on a point dataset and visualize and evaluate the clusters.

8. Implement hierarchical clustering on a dataset and plot the dendrogram.
9. Implement DBSCAN clustering on a dataset and visualize and evaluate the clusters.
10. Perform Principal Components Analysis (PCA) and apply any one or more classifiers
to show the performance variation with or without feature reduction.
11. Build a single layer perceptron model to classify AND, OR, and XOR problems (may
use TensorFlow/Keras) and visualize their decision boundaries. Also evaluate its performance.
12. Demonstrate the concept of boosting using the AdaBoost algorithm.

BCA 8014: System Programming using UNIX/Linux(Theory)

Course Learning Outcomes:

By the end of this course, a student should be able to:

- **Navigate and Automate** the UNIX/Linux environment using advanced command-line tools and complex shell scripts.
- **Develop System-Level Applications** in C that interact directly with the Linux Kernel via system calls.
- **Manage Lifecycle of Processes**, including creation, execution, and synchronization, while preventing common issues like "zombies" or "orphans."
- **Implement Inter-Process Communication (IPC)** to allow independent programs to share data and synchronize activities efficiently.
- **Analyze Memory Layouts** and manage dynamic memory while avoiding leaks, overflows, and race conditions.
- **Manipulate the File System** at a low level, understanding how inodes, file descriptors, and permissions function under the hood.

Credit: 03 **45 Hours**

Unit I: Introduction to UNIX/Linux (4 Hours)

History and philosophy of UNIX, GNU and Linux evolution, UNIX architecture (kernel, shell, system utilities), user space and kernel space, system call interface, Linux file system hierarchy, basic UNIX commands (ls, cp, mv, rm, cat, grep, wc,

awk etc.), I/O redirection and pipes, file permissions and ownership, environment variables, types of shells, shell initialization files.

Unit II: UNIX Shell Programming (10 Hours)

Introduction to shell scripting, structure of shell script, execution permissions, variables and environment variables, command substitution, arithmetic operations (expr, let, bc), conditional statements (if, if-else, case), looping constructs (for, while, until), break and continue, positional parameters, shift command, functions in shell, arrays, string operations, test command and file testing, input/output in scripts, trap and signal handling in shell, writing practical automation scripts.

Unit III: C Programming for System Programming (5 Hours)

Compilation process (preprocessing, compilation, linking), GCC compiler options, structure of C program in Linux, memory layout of C program (text, data, bss, heap, stack), pointers and dynamic memory allocation, file handling in C, low-level vs high-level I/O, static and shared libraries, Makefile basics, debugging using gdb, memory debugging using valgrind.

Unit IV: System Calls and File Management (5 Hours)

Concept of system calls, kernel mode switching, file descriptors, open(), close(), read(), write(), lseek(), stat(), fstat(), dup(), dup2(), directory handling (opendir(), readdir(), closedir()), file and directory permissions, chmod(), chown(), umask().

Unit V: Process Management (5 Hours)

Concept of process and process states, process control block, fork(), exec() family, wait() and waitpid(), exit() and _exit(),process scheduling basics, process identifiers (getpid(), getppid())

Unit VI: Inter-Process Communication (IPC) (5 Hours)

Need for IPC, unnamed pipes, named pipes (FIFO), message queues (System V and POSIX), shared memory, semaphores, synchronization between processes, introduction to sockets, client-server model basics.

Unit VII: Threads and Memory Management (5 Hours)

Processes vs threads, POSIX threads (pthread_create, pthread_join), thread synchronization using mutex, condition variables, race conditions and deadlocks, virtual memory concept, paging and segmentation, dynamic memory allocation (malloc(), free(), brk(), sbrk()), memory leaks and buffer overflow.

Unit VIII: File Systems, Security and Advanced Topics (6 Hours)

File system structure and inodes, hard and symbolic links, ext4 overview, mounting and unmounting file systems, file security and access control.

Reference Books

1. Advanced Programming in the UNIX Environment — W. Richard Stevens & Stephen A. Rago
2. *"Unix system administration" Maxwell – TMH.*
3. Sumitabha, Das, Unix Concepts And Applications, Tata McGraw-Hill Education, 2006
4. Michael Jang RHCSA/ RHCE Red Hat Linux Certification: Exams (Ex200 & Ex300) (Certification Press), 2011
5. Nemeth Synder & Hein, Linux Administration Handbook, Pearson Education, 2nd Edition ,2010
6.W. Richard Stevens, Bill Fenner, Andrew M. Rudoff, Unix Network Programming, The sockets Networking API, Vol. 1, 3rd Edition,2014

BCA 8014: System Programming using UNIX/Linux(Practical)

Credit: 01 **30 Hours**

1. Write a shell script to check if the number entered at the command line is prime or not.
2. Write a shell script to modify ―cal‖ command to display calendars of the specified months.
3. Write a shell script to modify ―cal‖ command to display calendars of the specified range of months.
4. Write a shell script to accept a login name. If not a valid login name display message – Entered login name is invalid‖.
5. Write a shell script to display date in the mm/dd/yy format.
6. Write a shell script to display on the screen sorted output of ―who‖ command along with the total number of users.
7. Write a shell script to compare two files and if found equal asks the user to delete the duplicate file.
8. Write a shell script to find the sum of digits of a given number.
9. Write a shell script to merge the contents of three files, sort the contents and then display them page by page.
10.Develop UNIX shell scripts using variables, control structures, and functions for automation tasks.
11.Write C programs using system calls for file handling and low-level I/O operations.
12.Implement process management concepts using fork(), exec(), wait(), and signals.
13.Design concurrent and IPC-based programs using pipes, shared memory, semaphores, and threads.

BCA 8021: Operations Research (Theory)

Course Learning Outcomes:

- Explain the historical development, nature, and scope of Operations Research.
- Formulate complex business scenarios into mathematical Linear Programming Models and solve them using **Graphical** and **Simplex** methods to find optimal solutions.
- Apply **Transportation** and **Assignment** algorithms to minimize costs and maximize efficiency in supply chain and resource allocation scenarios.
- Analyze competitive situations using **Game Theory** and apply **Decision Theory**
- Construct and analyze project networks using **PERT** and **CPM** .

Credit: 03 **45 Hours**

UNIT – I: Introduction: Historical Development: Definitions of OR; Nature and scope of Study; Phases of OR; Classification of OR models; Methodology of OR. **5 LH**

UNIT – II: Linear programming: Assumptions, Basic concepts; LP Formulation Graphical Solution – Feasible Region, Optimum Solution, special cases (unbounded solutions, infeasible solution and Alternative optima). **12 LH**

UNIT - III: Standard Form, Canonical form and Simplex method – maximization case, minimization case; Big – M method. **8 LH**

UNIT- IV: Transportation Problem: Transportation tableau, Mathematical Form, Methods for Finding Initial basic Feasible Solution – North West Corner Rule, Least Cost Method, VAM; Test for Optimality – MODI Method. **10 LH**

UNIT- V: Assignment Problem: Mathematical Statement of Problem, Comparison with Transportation Problem; Solution of Assignment Problem – Hungarian Methods. **10 LH**

UNIT- VI: Game Theory: Introduction; Two – person zero – some Games; Games with saddle Point – Pure Strategies; Rules of Dominances. **8 LH**

UNIT- VII: Decision Theory: Steps in Decision Theory Approach; Types of Decision-Making Environments; Decision Making Under Risk – Expected Monetary Value, Expected Opportunity loss; Decision Making Under Uncertainty – Criterion of optimism, Criterion of Pessimism, Laplace Criterion, Criterion on Regret. **8 LH**

UNIT- VIII: Network Analysis (PERT and CPM) : Introduction; Network Construction; Critical Path Analysis – Forward Pass, Backward Pass and Float of an Activity and Event, Critical Path, Estimation of Project Completion Time. **10 LH**

References Book :

1. Operations Research; Theory and Applications – J. K. Sharma; Macmillan.
2. Operation Research – P. K. Gupta and D. S. Hira; S. Chand.
3. Quantitative Technique in Management – N. D. Vhora; Tata McGraw Hill.
4. Operation Research – K. Swarup, P. K. Gupta and M. Mohan; Sultan Chand.
5. Operation Research: An introduction – H. A. Taha; Macmillan.

BCA 8021: Operations Research (Tutorial)

Credit:1 **15 Hour**

Tutorial based on concepts listed in theory.