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 Assess ment
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**			

Sem ester	Course Type	Name of the Course	Credit	Lect.	Tuto	Pract /Viva	Full Marks	Distribution of Marks		
								Theor y	Pract/ tuto./ Viva-voce	Internal Assess ment
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 …. 3021	Intermediate Level Course (Voc. Edn. & Trng.)	4				75			15
	Multi/Interdis ciplinary 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 of UGC recognized others	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 – 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

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

Introduction to worksheet and shell – getting started with Excel – Editing cell & using Commands and functions – Moving &Copying , Inserting & Deleting Rows & Columns - Printing work sheet. 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. **20hours**

Overview of Power point – presenting shows for corporate and commercial using Power point –Introduction to Desktop publishing – Computer viruses – Introduction to Internet – Web features. **15hours**

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**

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